PAT-NO: JP02000222338A

DOCUMENT-IDENTIFIER: JP 2000222338 A

TITLE: PERIPHERAL DEVICE, METHOD AND SYSTEM FOR PERIPHERAL DEVICE CONTROL, STORAGE MEDIUM STORED WITH PERIPHERAL DEVICE CONTROL PROGRAM, SENDING-OUT DEVICE SENDING OUT PERIPHERAL DEVICE CONTROL PROGRAM, AND PERIPHERAL DEVICE CONTROL PROGRAM PRODUCT, AND INFORMATION PROCESSOR, INFORMATION PROCESSING METHOD, STORAGE MEDIUM STORED WITH INFORMATION PROCESSING PROGRAM, SENDING-OUT DEVICE SENDING OUT INFORMATION PROCESSING PROGRAM, AND INFORMATION PROCESSING PROGRAM PRODUCT

PUBN-DATE: August 11, 2000

INVENTOR-INFORMATION:

NAME	COUNTRY
MATSUO, TAKUYUKI	N/A
ENDO, TOMOAKI	N/A
SASAKI, YASUHIKO	N/A
OSADA, MAMORU	N/A
INOUE, TAKASHI	N/A
SHIMODAIRA, MASAKO	N/A
TAKAGI, TOMOKO	N/A

ASSIGNEE-INFORMATION:

NAME	COUNTRY
CANON INC	N/A

APPL-NO: JP11285458

APPL-DATE: October 6, 1999

PRIORITY-DATA: 10333777 (November 25, 1998)

INT-CL (IPC): G06F013/12, B41J029/38 , G06F003/12

ABSTRACT:

PROBLEM TO BE SOLVED: To improve usability by inputting an interruption job from an information processor while a job is processed, and selecting and executing a device control program different from a device control program which is being executed.

SOLUTION: A printer driver connected to the information processor displays a dialog 5000 on a display at the time of issuing an interruption jog and in this dialog 5000, what is made to preferentially initiate interruption as an option of interruptive printing is set. After one option that a user desires is selected and completed from 5005 to 5007, the interruption jog is issued by pressing a print execution button 5001. This interruption job is inputted from the information processor while one of device control programs is executed to process the jog and a device control program different from the device control program which is being executed is selected and executed.

COPYRIGHT: (C)2000,JPO

(19)日本国特許庁（ＪＰ）　　(12) **公 開 特 許 公 報**（Ａ）

(11)特許出願公開番号
特開2000－222338
（P2000－222338A）

(43)公開日　平成12年８月11日(2000. 8. 11)

(51)Int.Cl.^{7}	識別記号	ＦＩ	テーマコード*（参考）
Ｇ０６Ｆ　13/12	３４０	Ｇ０６Ｆ　13/12　　３４０Ｈ	
		３４０Ｄ	
Ｂ４１Ｊ　29/38		Ｂ４１Ｊ　29/38　　Ｚ	
Ｇ０６Ｆ　3/12		Ｇ０６Ｆ　3/12　　Ｄ	

審査請求　未請求　請求項の数93　ＯＬ　（全 62 頁）

(21)出願番号　特願平11－285458
(22)出願日　平成11年10月６日(1999. 10. 6)
(31)優先権主張番号　特願平10－333777
(32)優先日　平成10年11月25日(1998. 11. 25)
(33)優先権主張国　日本（ＪＰ）

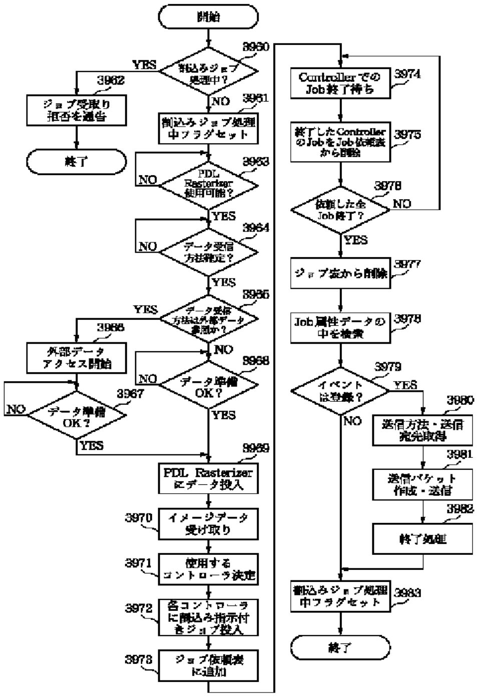
(71)出願人　000001007
キヤノン株式会社
東京都大田区下丸子３丁目30番２号
(72)発明者　松尾　卓幸
東京都大田区下丸子３丁目30番２号キヤノン株式会社内
(72)発明者　遠藤　友章
東京都大田区下丸子３丁目30番２号キヤノン株式会社内
(74)代理人　100090538
弁理士　西山　恵三　（外２名）

最終頁に続く

(54)【発明の名称】　**周辺機器及び周辺機器制御方法及び周辺機器制御システム及び周辺機器制御プログラムを記憶した記憶媒体及び周辺機器制御プログラムを送出する送出装置及び周辺機器制御プログラム製品及**

(57)【要約】
【課題】　ユーザーにとって使い勝手の良い、多機能な周辺機器を提供することができる。
【解決手段】　情報処理装置と接続された周辺機器で、周辺機器の複数のデバイスエンジンの機能を保持し当該デバイスエンジンでのジョブを管理する複数のデバイス制御プログラムの一つを実行してジョブを処理している間に、割り込みジョブが情報処理装置から入力されると、実行しているデバイス制御プログラムとは異なる他のデバイス制御プログラムを選択して実行することができる。

割り込みプリントジョブマネージャでのジョブ処理



【特許請求の範囲】

【請求項１】　情報処理装置と接続された周辺機器であって、

前記周辺機器の複数のデバイスエンジンの機能を保持し当該デバイスエンジンでのジョブを管理する複数のデバイス制御プログラムと、

前記複数のデバイス制御プログラムの一つを実行してジョブを処理している間に、割り込みジョブが前記情報処理装置から入力されると、前記実行しているデバイス制御プログラムとは異なる他のデバイス制御プログラムを選択して実行する制御手段とを有することを特徴とする周辺機器。

【請求項２】　前記デバイス制御プログラムは、前記割り込みジョブが入力された際に実行していたジョブの実行コンテキストを退避し、該割り込みジョブの実行が終了すると、前記退避したジョブの実行コンテキストを復帰することを特徴とする請求項１に記載の周辺機器。

【請求項３】　制御手段は、割り込みジョブ実行中は、他の割り込みジョブを多重に入力させないようにすることを特徴とする請求項１に記載の周辺機器。

【請求項４】　前記割り込みジョブは、割り込みプリントジョブであることを特徴とする請求項１に記載の周辺機器。

【請求項５】　前記割り込みジョブは、割り込みコピージョブであることを特徴とする請求項１に記載の周辺機器。

【請求項６】　前記制御手段は、前記割り込みコピージョブが入力された場合には、前記周辺機器のスキャナエンジンの機能を保持し当該スキャンエンジンでのジョブを管理するスキャナ制御プログラムにジョブが投入された後に、前記周辺機器のプリンタエンジンの機能を保持し当該プリンタエンジンでのジョブを管理する任意のプリンタ制御プログラムを選択することを特徴とする請求項５に記載の周辺機器。

【請求項７】　前記デバイス制御プログラムは、前記周辺機器のスキャナエンジンを制御するスキャナ制御プログラムであることを特徴とする請求項１に記載の周辺機器。

【請求項８】　前記デバイス制御プログラムは、前記周辺機器のレーザービームプリンタエンジンを制御するレーザービームプリンタ制御プログラムであることを特徴とする請求項１に記載の周辺機器。

【請求項９】　前記デバイス制御プログラムは、前記周辺機器のインクジェットプリンタエンジンを制御するインクジェットプリンタ制御プログラムであることを特徴とする請求項１に記載の周辺機器。

【請求項１０】　前記周辺機器は、前記デバイス制御プログラムを制御するプリントジョブ制御プログラムとコピージョブ制御プログラムとスキャンジョブ制御プログラムを有することを特徴とする請求項１に記載の周辺機器。

【請求項１１】　情報処理装置と接続された周辺機器における周辺機器制御方法であって、

前記周辺機器の複数のデバイスエンジンの機能を保持し当該デバイスエンジンでのジョブを管理する複数のデバイス制御プログラムの一つを実行してジョブを処理している間に、割り込みジョブが前記情報処理装置から入力されると、前記実行しているデバイス制御プログラムとは異なる他のデバイス制御プログラムを選択して実行することを特徴とする周辺機器制御方法。

【請求項１２】　前記デバイス制御プログラムは、前記割り込みジョブが入力された際に実行していたジョブの実行コンテキストを退避し、該割り込みジョブの実行が終了すると、前記退避したジョブの実行コンテキストを復帰することを特徴とする請求項１１に記載の周辺機器制御方法。

【請求項１３】　前記割り込みジョブ実行中は、他の割り込みジョブを多重に入力させないようにすることを特徴とする請求項１１に記載の周辺機器制御方法。

【請求項１４】　前記割り込みジョブは、割り込みプリントジョブであることを特徴とする請求項１１に記載の周辺機器制御方法。

【請求項１５】　前記割り込みジョブは、割り込みコピージョブであることを特徴とする請求項１１に記載の周辺機器制御方法。

【請求項１６】　前記割り込みコピージョブが入力された場合には、前記周辺機器のスキャナエンジンの機能を保持し当該スキャンエンジンでのジョブを管理するスキャナ制御プログラムにジョブが投入された後に、前記周辺機器のプリンタエンジンの機能を保持し当該プリンタエンジンでのジョブを管理する任意のプリンタ制御プログラムを選択することを特徴とする請求項１５に記載の周辺機器制御方法。

【請求項１７】　前記デバイス制御プログラムは、前記周辺機器のスキャナエンジンを制御するスキャナ制御プログラムであることを特徴とする請求項１１に記載の周辺機器制御方法。

【請求項１８】　前記デバイス制御プログラムは、前記周辺機器のレーザービームプリンタエンジンを制御するレーザービームプリンタ制御プログラムであることを特徴とする請求項１１に記載の周辺機器制御方法。

【請求項１９】　前記デバイス制御プログラムは、前記周辺機器のインクジェットプリンタエンジンを制御するインクジェットプリンタ制御プログラムであることを特徴とする請求項１１に記載の周辺機器制御方法。

【請求項２０】　前記周辺機器は、前記デバイス制御プログラムを制御するプリントジョブ制御プログラムとコピージョブ制御プログラムとスキャンジョブ制御プログラムを有することを特徴とする請求項１１に記載の周辺機器制御方法。

【請求項２１】　情報処理装置と接続された周辺機器で実行される周辺機器制御プログラムを記憶した記憶媒体であって、
前記周辺機器の複数のデバイスエンジンの機能を保持し当該デバイスエンジンでのジョブを管理する複数のデバイス制御プログラムの一つを実行してジョブを処理している間に、割り込みジョブが前記情報処理装置から入力されると、前記実行しているデバイス制御プログラムとは異なる他のデバイス制御プログラムを選択して実行する周辺機器制御プログラムを記憶することを特徴とする記憶媒体。

【請求項２２】　前記デバイス制御プログラムは、前記割り込みジョブが入力された際に実行していたジョブの実行コンテキストを退避し、該割り込みジョブの実行が終了すると、前記退避したジョブの実行コンテキストを復帰することを特徴とする請求項２１に記載の記憶媒体。

【請求項２３】　前記割り込みジョブ実行中は、他の割り込みジョブを多重に入力させないようにすることを特徴とする請求項２１に記載の記憶媒体。

【請求項２４】　前記割り込みジョブは、割り込みプリントジョブであることを特徴とする請求項２１に記載の記憶媒体。

【請求項２５】　前記割り込みジョブは、割り込みコピージョブであることを特徴とする請求項２１に記載の記憶媒体。

【請求項２６】　前記割り込みコピージョブが入力された場合には、前記周辺機器のスキャナエンジンの機能を保持し当該スキャンエンジンでのジョブを管理するスキャナ制御プログラムにジョブが投入された後に、前記周辺機器のプリンタエンジンの機能を保持し当該プリンタエンジンでのジョブを管理する任意のプリンタ制御プログラムを選択することを特徴とする請求項２５に記載の記憶媒体。

【請求項２７】　前記デバイス制御プログラムは、前記周辺機器のスキャナエンジンを制御するスキャナ制御プログラムであることを特徴とする請求項２１に記載の記憶媒体。

【請求項２８】　前記デバイス制御プログラムは、前記周辺機器のレーザービームプリンタエンジンを制御するレーザービームプリンタ制御プログラムであることを特徴とする請求項２１に記載の記憶媒体。

【請求項２９】　前記デバイス制御プログラムは、前記周辺機器のインクジェットプリンタエンジンを制御するインクジェットプリンタ制御プログラムであることを特徴とする請求項２１に記載の記憶媒体。

【請求項３０】　前記周辺機器は、前記デバイス制御プログラムを制御するプリントジョブ制御プログラムとコピージョブ制御プログラムとスキャンジョブ制御プログラムを有することを特徴とする請求項２１に記載の記憶媒体。

【請求項３１】　情報処理装置と周辺機器から構成される周辺機器制御システムであって、
前記周辺機器へジョブを出力する出力手段と、
前記周辺機器の複数のデバイスエンジンの機能を保持し当該デバイスエンジンでのジョブを管理する複数のデバイス制御プログラムと、
前記複数のデバイス制御プログラムの一つを実行してジョブを処理している間に、前記出力手段から割り込みジョブが前記情報処理装置へ出力されると、前記実行しているデバイス制御プログラムとは異なる他のデバイス制御プログラムを選択して実行する制御手段とを有することを特徴とする周辺機器制御システム。

【請求項３２】　前記デバイス制御プログラムは、前記割り込みジョブが入力された際に実行していたジョブの実行コンテキストを退避し、該割り込みジョブの実行が終了すると、前記退避したジョブの実行コンテキストを復帰することを特徴とする請求項３１に記載の周辺機器制御システム。

【請求項３３】　制御手段は、割り込みジョブ実行中は、他の割り込みジョブを多重に入力させないようにすることを特徴とする請求項３１に記載の周辺機器制御システム。

【請求項３４】　前記割り込みジョブは、割り込みプリントジョブであることを特徴とする請求項３１に記載の周辺機器制御システム。

【請求項３５】　前記割り込みジョブは、割り込みコピージョブであることを特徴とする請求項３１に記載の周辺機器制御システム。

【請求項３６】　前記制御手段は、前記割り込みコピージョブが入力された場合には、前記周辺機器のスキャナエンジンの機能を保持し当該スキャンエンジンでのジョブを管理するスキャナ制御プログラムにジョブが投入された後に、前記周辺機器のプリンタエンジンの機能を保持し当該プリンタエンジンでのジョブを管理する任意のプリンタ制御プログラムを選択することを特徴とする請求項３５に記載の周辺機器制御システム。

【請求項３７】　前記デバイス制御プログラムは、前記周辺機器のスキャナエンジンを制御するスキャナ制御プログラムであることを特徴とする請求項３１に記載の周辺機器制御システム。

【請求項３８】　前記デバイス制御プログラムは、前記周辺機器のレーザービームプリンタエンジンを制御するレーザービームプリンタ制御プログラムであることを特徴とする請求項３１に記載の周辺機器制御システム。

【請求項３９】　前記デバイス制御プログラムは、前記周辺機器のインクジェットプリンタエンジンを制御するインクジェットプリンタ制御プログラムであることを特徴とする請求項３１に記載の周辺機器制御システム。

【請求項４０】　前記周辺機器は、前記デバイス制御プ

ログラムを制御するプリントジョブ制御プログラムとコピージョブ制御プログラムとスキャンジョブ制御プログラムを有することを特徴とする請求項３１に記載の周辺機器制御システム。

【請求項４１】　情報処理装置と接続された周辺機器で実行される周辺機器制御プログラムを送出する送出装置であって、

前記周辺機器の複数のデバイスエンジンの機能を保持し当該デバイスエンジンでのジョブを管理する複数のデバイス制御プログラムの一つを実行してジョブを処理している間に、割り込みジョブが前記情報処理装置から入力されると、前記実行しているデバイス制御プログラムとは異なる他のデバイス制御プログラムを選択して実行する周辺機器制御プログラムを送出することを特徴とする送出装置。

【請求項４２】　前記デバイス制御プログラムは、前記割り込みジョブが入力された際に実行していたジョブの実行コンテキストを退避し、該割り込みジョブの実行が終了すると、前記退避したジョブの実行コンテキストを復帰することを特徴とする請求項４１に記載の送出装置。

【請求項４３】　前記割り込みジョブ実行中は、他の割り込みジョブを多重に入力させないようにすることを特徴とする請求項４１に記載の送出装置。

【請求項４４】　前記割り込みジョブは、割り込みプリントジョブであることを特徴とする請求項４１に記載の送出装置。

【請求項４５】　前記割り込みジョブは、割り込みコピージョブであることを特徴とする請求項４１に記載の送出装置。

【請求項４６】　前記割り込みコピージョブが入力された場合には、前記周辺機器のスキャナエンジンの機能を保持し当該スキャンエンジンでのジョブを管理するスキャナ制御プログラムにジョブが投入された後に、前記周辺機器のプリンタエンジンの機能を保持し当該プリンタエンジンでのジョブを管理する任意のプリンタ制御プログラムを選択することを特徴とする請求項４５に記載の送出装置。

【請求項４７】　前記デバイス制御プログラムは、前記周辺機器のスキャナエンジンを制御するスキャナ制御プログラムであることを特徴とする請求項４１に記載の送出装置。

【請求項４８】　前記デバイス制御プログラムは、前記周辺機器のレーザービームプリンタエンジンを制御するレーザービームプリンタ制御プログラムであることを特徴とする請求項４１に記載の送出装置。

【請求項４９】　前記デバイス制御プログラムは、前記周辺機器のインクジェットプリンタエンジンを制御するインクジェットプリンタ制御プログラムであることを特徴とする請求項４１に記載の送出装置。

【請求項５０】　前記周辺機器は、前記デバイス制御プログラムを制御するプリントジョブ制御プログラムとコピージョブ制御プログラムとスキャンジョブ制御プログラムを有することを特徴とする請求項４１に記載の送出装置。

【請求項５１】　情報処理装置と接続された周辺機器で実行される周辺機器制御プログラム製品であって、

前記周辺機器の複数のデバイスエンジンの機能を保持し当該デバイスエンジンでのジョブを管理する複数のデバイス制御プログラムの一つを実行してジョブを処理している間に、割り込みジョブが前記情報処理装置から入力されると、前記実行しているデバイス制御プログラムとは異なる他のデバイス制御プログラムを選択して実行することを特徴とする周辺機器制御プログラム製品。

【請求項５２】　前記デバイス制御プログラムは、前記割り込みジョブが入力された際に実行していたジョブの実行コンテキストを退避し、該割り込みジョブの実行が終了すると、前記退避したジョブの実行コンテキストを復帰することを特徴とする請求項５１に記載の周辺機器制御プログラム製品。

【請求項５３】　前記割り込みジョブ実行中は、他の割り込みジョブを多重に入力させないようにすることを特徴とする請求項５１に記載の周辺機器制御プログラム製品。

【請求項５４】　前記割り込みジョブは、割り込みプリントジョブであることを特徴とする請求項５１に記載の周辺機器制御プログラム製品。

【請求項５５】　前記割り込みジョブは、割り込みコピージョブであることを特徴とする請求項５１に記載の周辺機器制御プログラム製品。

【請求項５６】　前記割り込みコピージョブが入力された場合には、前記周辺機器のスキャナエンジンの機能を保持し当該スキャンエンジンでのジョブを管理するスキャナ制御プログラムにジョブが投入された後に、前記周辺機器のプリンタエンジンの機能を保持し当該プリンタエンジンでのジョブを管理する任意のプリンタ制御プログラムを選択することを特徴とする請求項５５に記載の周辺機器制御プログラム製品。

【請求項５７】　前記デバイス制御プログラムは、前記周辺機器のスキャナエンジンを制御するスキャナ制御プログラムであることを特徴とする請求項５１に記載の周辺機器制御プログラム製品。

【請求項５８】　前記デバイス制御プログラムは、前記周辺機器のレーザービームプリンタエンジンを制御するレーザービームプリンタ制御プログラムであることを特徴とする請求項５１に記載の周辺機器制御プログラム製品。

【請求項５９】　前記デバイス制御プログラムは、前記周辺機器のインクジェットプリンタエンジンを制御するインクジェットプリンタ制御プログラムであることを特

徴とする請求項５１に記載の周辺機器制御プログラム製品。

【請求項６０】　前記周辺機器は、前記デバイス制御プログラムを制御するプリントジョブ制御プログラムとコピージョブ制御プログラムとスキャンジョブ制御プログラムを有することを特徴とする請求項５１に記載の周辺機器制御プログラム製品。

【請求項６１】　前記デバイス制御プログラムを識別するための識別情報を表示する表示手段を有することを特徴とする請求項３１に記載の周辺機器制御システム。

【請求項６２】　前記表示手段は、前記デバイス制御プログラムをアイコンで表現することを特徴とする請求項６１に記載の周辺機器制御システム。

【請求項６３】　前記表示手段は、割込みジョブが出力されたときに選択、実行される前記デバイス制御プログラムを表現するアイコンを表示することを特徴とする請求項６２に記載の周辺機器制御システム。

【請求項６４】　前記表示手段は、割込みジョブが出力されたときに選択、実行される前記デバイス制御プログラムが管理するデバイスエンジンの情報を表示することを特徴とする請求項６１の周辺機器制御システム。

【請求項６５】　前記表示手段は、割込みジョブが出力されたときに選択、実行される前記デバイス制御プログラムが有する機能を表示することを特徴とする請求項６１の周辺機器システム。

【請求項６６】　前記表示手段は、前記複数のデバイス制御プログラムを表現するアイコンをリスト表示すると共に、割込みジョブが出力されたときに選択、実行される前記デバイス制御プログラムのアイコンが他の前記デバイス制御プログラムのアイコンと識別されるように表示することを特徴とする請求項６２に記載の周辺機器制御システム。

【請求項６７】　前記表示手段は、割込みジョブが指示されたときに、割込みジョブが出力されたときに選択、実行されうる前記デバイス制御プログラムを識別する情報が複数、リスト表示されることを特徴とする請求項６１に記載の周辺機器制御システム。

【請求項６８】　前記表示手段により表示されている識別情報のいずれかを選択するための選択手段を有し、前出力手段記は、前記選択手段により選択された識別情報に対応するデバイス制御プログラムを指示したジョブを出力することを特徴とする請求項６１に記載の周辺機器制御システム。

【請求項６９】　周辺機器へジョブを出力する情報処理装置であって、前記周辺機器におけるデバイスエンジンの機能を保持し当該デバイスエンジンでのジョブを管理するデバイス制御プログラムに関する情報を取得する取得手段と、前記取得手段により取得された情報に基づいて、前記デバイス制御プログラムを識別する識別情報を表示する表示手段と、前記表示手段により表示された識別情報を選択するための選択手段と、前記選択手段により選択された識別情報に対応するデバイス制御プログラムを指示したジョブを出力する出力手段とを有することを特徴とする情報処理装置。

【請求項７０】　前記表示手段は、割込みジョブが出力されたときに選択、実行されるデバイス制御プログラムを識別する識別情報を表示することを特徴とする請求項６９に記載の情報処理装置。

【請求項７１】　前記表示手段は、割込みジョブが出力されたときに選択、実行される前記デバイス制御プログラムが管理するデバイスエンジンの情報を表示することを特徴とする請求項６９の情報処理装置。

【請求項７２】　前記表示手段は、割込みジョブが出力されたときに選択、実行される前記デバイス制御プログラムが有する機能を表示することを特徴とする請求項６９の情報処理装置。

【請求項７３】　前記表示手段は、前記複数のデバイス制御プログラムを表現するアイコンをリスト表示すると共に、割込みジョブが出力されたときに選択、実行される前記デバイス制御プログラムのアイコンが他の前記デバイス制御プログラムのアイコンと識別されるように表示することを特徴とする請求項６９に記載の情報処理装置。

【請求項７４】　周辺機器へジョブを出力する情報処理方法であって、前記周辺機器におけるデバイスエンジンの機能を保持し当該デバイスエンジンでのジョブを管理するデバイス制御プログラムに関する情報を取得する取得ステップと、前記取得手段により取得された情報に基づいて、前記デバイス制御プログラムを識別する識別情報を表示手段に表示する表示ステップと、前記表示手段により表示された識別情報を選択するための選択ステップと、前記選択手段により選択された識別情報に対応するデバイス制御プログラムを指示したジョブを出力する出力ステップとを有することを特徴とする情報処理方法。

【請求項７５】　前記表示ステップでは、割込みジョブが出力されたときに選択、実行されるデバイス制御プログラムを識別する識別情報を表示することを特徴とする請求項７４に記載の情報処理方法。

【請求項７６】　前記表示ステップでは、割込みジョブが出力されたときに選択、実行される前記デバイス制御プログラムが管理するデバイスエンジンの情報を表示することを特徴とする請求項７４の情報処理方法。

【請求項７７】　前記表示ステップでは、割込みジョブが出力されたときに選択、実行される前記デバイス制御プログラムが有する機能を表示することを特徴とする請求項７４の情報処理方法。

【請求項７８】　前記表示ステップでは、前記複数のデバイス制御プログラムを表現するアイコンをリスト表示すると共に、割込みジョブが出力されたときに選択、実

行される前記デバイス制御プログラムのアイコンが他の前記デバイス制御プログラムのアイコンと識別されるように表示することを特徴とする請求項７４に記載の情報処理方法。

【請求項７９】　周辺機器へジョブを出力する情報処理プログラムを格納した、コンピュータで読取可能な記録媒体であって、当該情報処理プログラムは、前記周辺機器におけるデバイスエンジンの機能を保持し当該デバイスエンジンでのジョブを管理するデバイス制御プログラムに関する情報を取得する取得ステップと、前記取得手段により取得された情報に基づいて、前記デバイス制御プログラムを識別する識別情報を表示手段に表示する表示ステップと、前記表示手段により表示された識別情報を選択するための選択ステップと、前記選択手段により選択された識別情報に対応するデバイス制御プログラムを指示したジョブを出力する出力ステップとを有することを特徴とする記録媒体。

【請求項８０】　前記表示ステップでは、割込みジョブが出力されたときに選択、実行されるデバイス制御プログラムを識別する識別情報を表示することを特徴とする請求項７９に記載の記録媒体。

【請求項８１】　前記表示ステップでは、割込みジョブが出力されたときに選択、実行される前記デバイス制御プログラムが管理するデバイスエンジンの情報を表示することを特徴とする請求項７９の記録媒体。

【請求項８２】　前記表示ステップでは、割込みジョブが出力されたときに選択、実行される前記デバイス制御プログラムが有する機能を表示することを特徴とする請求項７９の記録媒体。

【請求項８３】　前記表示ステップでは、前記複数のデバイス制御プログラムを表現するアイコンをリスト表示すると共に、割込みジョブが出力されたときに選択、実行される前記デバイス制御プログラムのアイコンが他の前記デバイス制御プログラムのアイコンと識別されるように表示することを特徴とする請求項７９に記載の記録媒体。

【請求項８４】　周辺機器へジョブを出力する情報処理プログラムを送出する送出装置であって、当該情報処理プログラムは、前記周辺機器におけるデバイスエンジンの機能を保持し当該デバイスエンジンでのジョブを管理するデバイス制御プログラムに関する情報を取得する取得ステップと、前記取得手段により取得された情報に基づいて、前記デバイス制御プログラムを識別する識別情報を表示手段に表示する表示ステップと、前記表示手段により表示された識別情報を選択するための選択ステップと、前記選択手段により選択された識別情報に対応するデバイス制御プログラムを指示したジョブを出力する出力ステップとを有することを特徴とする送出装置。

【請求項８５】　前記表示ステップでは、割込みジョブが出力されたときに選択、実行されるデバイス制御プログラムを識別する識別情報を表示することを特徴とする請求項８４に記載の送出装置。

【請求項８６】　前記表示ステップでは、割込みジョブが出力されたときに選択、実行される前記デバイス制御プログラムが管理するデバイスエンジンの情報を表示することを特徴とする請求項８４の送出装置。

【請求項８７】　前記表示ステップでは、割込みジョブが出力されたときに選択、実行される前記デバイス制御プログラムが有する機能を表示することを特徴とする請求項８４の送出装置。

【請求項８８】　前記表示ステップでは、前記複数のデバイス制御プログラムを表現するアイコンをリスト表示すると共に、割込みジョブが出力されたときに選択、実行される前記デバイス制御プログラムのアイコンが他の前記デバイス制御プログラムのアイコンと識別されるように表示することを特徴とする請求項８４に記載の送出装置。

【請求項８９】　周辺機器へジョブを出力する情報処理プログラム製品であって、当該情報処理プログラムは、前記周辺機器におけるデバイスエンジンの機能を保持し当該デバイスエンジンでのジョブを管理するデバイス制御プログラムに関する情報を取得する取得ステップと、前記取得手段により取得された情報に基づいて、前記デバイス制御プログラムを識別する識別情報を表示手段に表示する表示ステップと、前記表示手段により表示された識別情報を選択するための選択ステップと、前記選択手段により選択された識別情報に対応するデバイス制御プログラムを指示したジョブを出力する出力ステップとを有することを特徴とする情報処理プログラム製品。

【請求項９０】　前記表示ステップでは、割込みジョブが出力されたときに選択、実行されるデバイス制御プログラムを識別する識別情報を表示することを特徴とする請求項８９に記載の情報処理プログラム製品。

【請求項９１】　前記表示ステップでは、割込みジョブが出力されたときに選択、実行される前記デバイス制御プログラムが管理するデバイスエンジンの情報を表示することを特徴とする請求項８９の情報処理プログラム製品。

【請求項９２】　前記表示ステップでは、割込みジョブが出力されたときに選択、実行される前記デバイス制御プログラムが有する機能を表示することを特徴とする請求項８９の情報処理プログラム製品。

【請求項９３】　前記表示ステップでは、前記複数のデバイス制御プログラムを表現するアイコンをリスト表示すると共に、割込みジョブが出力されたときに選択、実行される前記デバイス制御プログラムのアイコンが他の前記デバイス制御プログラムのアイコンと識別されるように表示することを特徴とする請求項８９に記載の情報処理プログラム製品。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】本発明は、ネットワーク、IEEE 1394等の接続形態を介して情報処理装置から制御されるプリンタ、スキャナ、ファックス、コピー機等の周辺機器及び周辺機器制御方法及び周辺機器制御システム及び周辺機器制御プログラムを記憶した記憶媒体に関する。

【０００２】

【従来の技術】ネットワーク等に接続されたプリンタ、スキャナ、ファックス、コピー機に対して、ホストコンピューター（情報処理装置）からジョブを投入する場合には、大容量の記憶装置を搭載する機器では同時に多くのジョブを内部に蓄積することが可能となった。通常、ジョブの投入順に基づいたファーストインファーストサーブ（FIFS）方式でジョブが処理される。

【０００３】

【発明が解決しようとする課題】従来の周辺機器、特に、プリンタ、スキャナ、ファックス、コピー機の機能を一つの筐体に有する多機能周辺機器においては、割り込みジョブを制御することができなかった。

【０００４】本発明の目的は、ユーザーにとって使い勝手の良い、多機能な周辺機器及び周辺機器制御方法及び周辺機器制御システム及び周辺機器制御プログラムを記憶した記憶媒体を提供することにある。

【０００５】

【課題を解決するための手段】上記目的を達成するための本発明の周辺機器は、情報処理装置と接続された周辺機器であって、前記周辺機器の複数のデバイスエンジンの機能を保持し当該デバイスエンジンでのジョブを管理する複数のデバイス制御プログラムと、前記複数のデバイス制御プログラムの一つを実行してジョブを処理している間に、割り込みジョブが前記情報処理装置から入力されると、前記実行しているデバイス制御プログラムとは異なる他のデバイス制御プログラムを選択して実行する制御手段とを有する。

【０００６】上記目的を達成するための本発明の周辺機器制御方法は、情報処理装置と接続された周辺機器における周辺機器制御方法であって、前記周辺機器の複数のデバイスエンジンの機能を保持し当該デバイスエンジンでのジョブを管理する複数のデバイス制御プログラムの一つを実行してジョブを処理している間に、割り込みジョブが前記情報処理装置から入力されると、前記実行しているデバイス制御プログラムとは異なる他のデバイス制御プログラムを選択して実行する。

【０００７】上記目的を達成するための本発明の記憶媒体は、情報処理装置と接続された周辺機器で実行される周辺機器制御プログラムを記憶した記憶媒体であって、前記周辺機器の複数のデバイスエンジンの機能を保持し当該デバイスエンジンでのジョブを管理する複数のデバイス制御プログラムの一つを実行してジョブを処理している間に、割り込みジョブが前記情報処理装置から入力されると、前記実行しているデバイス制御プログラムとは異なる他のデバイス制御プログラムを選択して実行する周辺機器制御プログラムを記憶する。

【０００８】上記目的を達成するための本発明の周辺機器制御システムは、情報処理装置と周辺機器から構成される周辺機器制御システムであって、前記周辺機器へジョブを出力する出力手段と、前記周辺機器の複数のデバイスエンジンの機能を保持し当該デバイスエンジンでのジョブを管理する複数のデバイス制御プログラムと、前記複数のデバイス制御プログラムの一つを実行してジョブを処理している間に、前記出力手段から割り込みジョブが前記情報処理装置へ出力されると、前記実行しているデバイス制御プログラムとは異なる他のデバイス制御プログラムを選択して実行する制御手段とを有する。

【０００９】上記目的を達成するための本発明の送出装置は、情報処理装置と接続された周辺機器で実行される周辺機器制御プログラムを送出する送出装置であって、前記周辺機器の複数のデバイスエンジンの機能を保持し当該デバイスエンジンでのジョブを管理する複数のデバイス制御プログラムの一つを実行してジョブを処理している間に、割り込みジョブが前記情報処理装置から入力されると、前記実行しているデバイス制御プログラムとは異なる他のデバイス制御プログラムを選択して実行する周辺機器制御プログラムを送出する。

【００１０】上記目的を達成するための本発明の周辺機器制御プログラム製品は、情報処理装置と接続された周辺機器で実行される周辺機器制御プログラム製品であって、前記周辺機器の複数のデバイスエンジンの機能を保持し当該デバイスエンジンでのジョブを管理する複数のデバイス制御プログラムの一つを実行してジョブを処理している間に、割り込みジョブが前記情報処理装置から入力されると、前記実行しているデバイス制御プログラムとは異なる他のデバイス制御プログラムを選択して実行する。

【００１１】上記目的を接待するための本発明の情報処理装置は、周辺機器へジョブを出力する情報処理装置であって、前記周辺機器におけるデバイスエンジンの機能を保持し当該デバイスエンジンでのジョブを管理するデバイス制御プログラムに関する情報を取得し、取得された情報に基づいて、前記デバイス制御プログラム、特に、割込みジョブが出力されたときに選択、実行されるデバイス制御プログラムを識別する識別情報を表示し、選択された識別情報に対応するデバイス制御プログラムを指示したジョブを出力する。

【００１２】

【発明の実施の形態】（多機能周辺機器の構成）図1は、本発明の一実施形態を示す多機能周辺機器（MFP :Multi-Function Peripheral）の構成を示す図である。

【００１３】図1において、101は多機能周辺機器を制御

するためのコントローラであり、図3に示すハードウエア構成を有し、図76の記憶媒体としてのDISK315に示すようなソフトウエア（制御プログラム）構成を有している。102はスキャナエンジンであり、コントローラ101によって制御されている。103,104はレーザービームプリンタエンジン（LBP）1、2であり、コントローラ101によって制御されている。103のレーザービームプリンタエンジン1はフィニッシャー106に接続されており、同プリンタエンジンから出力された複数の記録媒体（例えば、紙）をまとめてステープル（ホッチキス）処理することが可能となっている。フィニッシャー106もコントローラ101によって制御されている。105はインクジェットプリントエンジン（IJP）でカラー印字も可能であり、コントローラ101によって制御されている。107はネットワーク（イーサーネット）インターフェースであり、コントローラ101に対して同インターフェースを通した双方向通信を提供している。108はIEEE1394インターフェースであり、コントローラ101に対して同インターフェースを通した双方向通信を提供している。109はIEEE1284インターフェースであり、コントローラ101に対して同インターフェースを通した双方向通信を提供している。110はユーザーインタフェースであり、LCDディスプレイとキーボードから構成されており、コントローラ101からの情報表示を行うとともにユーザからの指示をコントローラ101に伝える。

【００１４】以上のような構成を有する多機能周辺機器は、3つの物理プリンタ、LBP103(B/W, Finisher付), LBP104(B/W), IJP105(Color)を選択させてプリントジョブの発行を可能にする。また、4つの論理プリンタ（クラスタ）プリンタ、LBP103+LBP104, LBP104+IJP105, LBP103+IJP105, LBP103+LBP104+IJP105を選択させてプリントジョブの発行を可能にする。また、上記7つのプリンタ、LBP103, LBP104, IJP105, LBP103+LBP104, LBP104+IJP105, LBP103+IJP105, LBP103+LBP104+IJP105から自動選択させてプリントジョブの発行を可能にする。更に、IJPを使用し、白黒のみ印刷可能なプリンターとする論理プリンターを構成しプリントジョブの発行を可能にする。

【００１５】また、外部からスキャンジョブの発行が可能になり、カラー原稿の読込みが可能になる。また、スキャナーと上記8つのプリンタ、LBP103, LBP104, IJP105, LBP103+LBP104, LBP104+IJP105, LBP103+IJP105, LBP103+LBP104+IJP105白黒のみ印刷可能なIJPを選択（自動選択も可能）させてコピージョブの発行を可能にする。更に、プリンタがIJPのみの場合にはカラーコピージョブの発行を可能にする。

【００１６】また、フォント、フォームオーバーレイをプリントジョブで使用でき、アップロード・ダウンロードが可能でリソース管理ができる。また、カラープロファイルをプリントジョブ・スキャンジョブ・コピージョブで使用でき、アップロード・ダウンロードが可能でリソース管理ができる。更にログを機器内部で自動生成できアップロードが可能になる。

【００１７】また、Network(Ethernet, TCP/IP), IEEE1284, IEEE1394のどのインターフェースからでも、全ての機能を使用可能である。また、各インターフェースごとにサブアドレスと物理・論理デバイス（プリンター、スキャナ、コピー）および各リソースの対応が取られており、サブアドレスを選択することによりこれらの物理・論理デバイスおよび各リソースを選択することが可能になる。これらの各インターフェースの各サブアドレスに対して、同時にジョブを発行およびダウンロード・アップロードを指示することが可能になる。また、サブアドレスと物理・論理デバイスおよび各リソースとの対応は、機器自身に問い合わせることにより得られる。更に、内部で持つユーザーインターフェースからも、機能を使用可能になる。

【００１８】また、外部から上記インターフェースを通して後述するSupervisorに問い合わせることにより、提供している機能概要（どんなジョブが発行できるか、どんなリソースが利用できるかなど）、ジョブ発行、リソースのダウンロード・アップロードに使用するサブアドレス、各機能の詳細情報（最大コピー部数、フィニシャーの種類サポートしているPDL、指定できる出力BINの数等）を取得可能になる。上記情報を使用して、Client側ではソフトウエア（制御プログラム）を自動構成することも可能になる。ただし、この問合わせに使用するサブアドレスだけは各インターフェースにおいて、Clientはあらかじめ知っておく必要がある。管理者は、上記の各種機能を接続形態および使用ユーザにより制限可能になる。

【００１９】（全体システム構成）図2は本実施形態のシステム構成を示す図である。図2において、201は図1に示す多機能周辺機器（MFP）で、多機能周辺機器201はネットワークインタフェースケーブル208（10BASE-T）、IEEE1394インタフェースケーブル206、IEEE1284インタフェースケーブル207によってClientPC202、203、204、205（情報処理装置）にそれぞれ接続されている。図78に示すような各Client PCでは、図77に示す記憶媒体としての後述するDISK6009に記憶されている各種ソフトウエア（制御プログラム）が動作している。Ethernet208に接続されているClientPC202,203はIPアドレス、ポート番号を指定して多機能周辺機器201に接続しIPパケットデータを出力する。IEEE1394インタフェース206に接続されているClientPC204はノードIDとLUN（ロジカルユニットナンバー）を指定して多機能周辺機器201と接続しSBP−2パケットデータを出力する。IEEE1284インタフェース207に接続されているClientPC205はソケット番号を指定して多機能周辺機器201と接続しIEEE1284．4パケットデータを出力する。

【００２０】（コントローラのハードウエア構成）図3は、図1に示す本実施形態のコントローラ101のハードウエア構成を示す図である。コントローラ101の内部では、CPU301がバス313を介して、Memory(RAM)302,ユーザーインターフェース（操作部）110を構成するLCDディスプレイ303及びキーボード304,ROM314，DISK315が接続されている。図76に示す各種プログラム及びデータは、ハードディスクやフロッピーディスク等のDISK315（記憶媒体）に記憶されており、必要に応じて順次Memory(RAM)302に読み出されてCPU301で実行される。このDISK315は、MFP201に着脱可能でもMFP201に内蔵されたものでも良い。更に、図76に示すプログラムは、ネットワークインタフェースケーブル208（10BASE-T）、IEEE1394インタフェースケーブル206、 IEEE1284インタフェースケーブル207を介してClientPC202、203、204、205或いは他のMFPからダウンロードされてDISK315に記憶される構成でも良い。

【００２１】LCDディスプレイ303、キーボード304は図1に示すユーザインターフェース（操作部）110を構成するものであり、CPU301がLCDディスプレイ303にデータを書き込むことにより表示を行い、CPU301がキーボード304からデータを読み出すことにより、ユーザからの指示を入力する。

【００２２】また、バス313には、ネットワークインターフェースコネクタ305・IEEE1394インターフェースコネクタ306・IEEE1284インターフェースコネクタ307が接続されており、それぞれ図1に示すネットワークインターフェース107、IEEE1394インターフェース108、IEEE1284インターフェース109に対応し、図2に示すEthernet(10BASE-T)ケーブル208、IEEE1394ケーブル206、IEEE1284ケーブル207が接続される。CPU301がこれらのインターフェースからデータを読み込みまたは書き込むことによりそれぞれのインターフェースを使用した通信を行う。

【００２３】更に、バス313には、Laser Beam Printer Engine308，Finisher309,Scanner Engine310，Laser Beam Printer Engine311，Ink Jet Printer Engine312が接続されており、これらは、図1に示すLaser Beam Printer Engine103，Finisher106,Scanner Engine102，Laser Beam Printer Engine104，Ink Jet Printer Engine105に対応するものであり、CPU301がこれらのエンジンに対してデータの読み書きを行うことによりプリント・スキャンなどのエンジン動作および各種ステータス取得を行う。なお、Laser Beam Printer Engine308，Finisher309,Scanner Engine310，Laser Beam Printer Engine311，Ink Jet Printer Engine312は、MFP201の内部ではなく、ネットワーク上にそれぞれ単体の周辺機器として存在し、それをMFP201のコントローラー101が制御しても良い。

【００２４】（コントローラ101のソフトウエア構成）図4は、コントローラ101内のDISK315のソフトウエア（制御プログラム）構成を示す図であり、これらのソフトウエア（制御プログラム）はCPU301が実行する。図の中で、実線がデータ及び制御を示し、点線が設定及び能力の取得を示す。401はユーザーインターフェースドライバであり、LCDディスプレイ303およびキーボード304の制御を行う。405はユーザインターフェースマネージャ（制御プログラム）であり、UIドライバ401から入力されたユーザからの入力情報を解釈して、図46で示すコマンドパケットを生成しインタープリター（制御プログラム）409に出力する。ユーザーインターフェースマネージャ405はまたインタープリター409から入力されたコマンドパケットを解釈し、UIドライバ（制御プログラム）401を介してLCDディスプレイ303に表示を行う。

【００２５】402はネットワークインターフェースドライバ（制御プログラム）であり、ネットワークインターフェースコネクタ305の制御を行い、ネットワークパケットの物理層（物理パケット）の処理、つまり物理パケットからトランスポートパケットの抽出、およびトランスポートパケットから物理パケットの生成を行う。406はTCP／IP・UDP／IP処理モジュールであり、ネットワークインターフェース402から出力されたトランスポートパケットの処理をおこないコマンドパケットを抽出してインタープリター409に出力する。また、インタープリター409から出力されたコマンドパケットからトランスポートパケットの生成を行い、ネットワークインターフェース402に出力する。

【００２６】403はIEEE1284ドライバ（制御プログラム）であり、IEEE1284インターフェース307の制御を行う。407はIEEE1284.4の処理モジュール（制御プログラム）である。IEEE1284.4は主にIEEE1284インターフェースで使用されることを目的としたトランスポート層の標準である。IEEE1284.4の処理モジュール（制御プログラム）407はIEEE1284ドライバ403から出力されたトランスポートパケットの処理をおこないコマンドパケットを抽出してインタープリター409に出力する。また、インタープリター409から出力されたコマンドパケットからトランスポートパケットの生成を行い、IEEE1284ドライバ403に出力する。404はIEEE1394ドライバ（制御プログラム）であり、IEEE1394インターフェース306の制御を行う。

【００２７】408はIEEE1394におけるトランスポート層であるSBP（シリアルバスプロトコル）－2を処理するモジュール（制御プログラム）である。SBP－2の処理モジュール408はIEEE1394ドライバ404から出力されたトランスポートパケットの処理をおこないコマンドパケットを抽出してインタープリター409に出力する。また、インタープリター409から出力されたコマンドパケットからトランスポートパケットの生成を行い、IEEE1394ドライバ404に出力する。409は406・407・408の各トランスポート処理部およびユーザインターフェースマネージャー

405から入力されたコマンドパケットを解釈しコマンドを生成するパケットインタープリターである。パケットインタープリター409は他のモジュール（制御プログラム）からの依頼によりコマンドパケットの生成も行う。インタープリター409はSupervisor（統括管理マネージャ）410が有するサブアドレス：サービスID対応表（図9）に従って、有効となっているサブアドレスから入力されたコマンドパケットのみを処理し、それ以外のサブアドレスから入力されたコマンドパケットは破棄する。

【００２８】410は、図7（属性表）、図9（サブアドレス：サービスID対応表）、図11（サービスID：タスクタイプ対応表）、図13（ユーザー認証表）、図14（アクセス制御表）、図16（イベント設定表）、図17（イベントフォーマット表）で示す各種データをDISK315に保持し、コントローラ101の動作を統括管理するSupervisor（統括管理マネージャ）である。Supervisor410は、自分自身が保持する各種データ、各マネージャ（フォントマネージャ413、フォームオーバーレイマネージャ414、ログマネージャ415、カラープロファイルマネージャ416、プリントマネージャ501－509、スキャンジョブマネージャ419およびコピージョブマネージャ601－608）が保持する各種データ、および各コントローラ（プリンタコントローラ510－512、スキャナコントローラ420）が保持する各種データを、入力されたコマンドの指示に従って参照・変更する。

【００２９】411はセキュリティゲートであり、インタープリター409で解釈されたコマンドパケットをSupervisor410（制御プログラム）が有するユーザ認証表（図13）およびアクセス制御表（図14）に従って、正しい使用権を持つユーザのみが、多機能周辺機器201を使用可能とするべくコマンドの入力を制限するものである。412はディスパッチャー（制御プログラム）であり、インタープリター409から入力されたコマンドを、これらを処理するための制御プログラムである各リソースマネージャ（フォントマネージャ413、フォームオーバーレイマネージャ414、ログマネージャ415、カラープロファイルマネージャ416）と各ジョブマネージャー（プリントマネージャ501－509、スキャンジョブマネージャ419およびコピージョブマネージャ601－608）に配布する。413はフォントマネージャであり、フォントの管理を行う。フォントマネージャ413は図26（属性表）、図28（FONT表）で示すようなデータをDISK315に保持している。フォントマネージャ413で管理されるフォント（DISK315或いはROM314に記憶されている）は、PDLラスタライザ417および418に渡され、プリントジョブの処理の際に使用される。

【００３０】414はフォームオーバーレイマネージャであり、フォームオーバーレイの管理を行う。フォームオーバーレイマネージャ414は図29（属性表）、図31（FORM表）で示すようなデータをDISK315に保持している。フォームオーバーレイマネージャ414で管理されるフォームオーバーレイ（DISK315或いはROM314に記憶されている）は、PDLラスタライザ417および418に渡され、プリントジョブの処理の際に使用される。415はログマネージャであり、ログの管理を行う。ログマネージャ415は図32（属性表）、図34（LOG表）で示すようなデータをDISK315に保持している。ログは各マネージャから出力され、ログマネージャの管理下におかれる。実施のログファイルは、RAM302或いはDISK315に記憶され随時更新されている。

【００３１】416カラープロファイルマネージャであり、カラースキャナやカラープリンタがカラーマッチングを行うためのカラープロファイルデータの管理を行う。カラープロファイルマネージャ416は図37（属性表）、図39（カラープロファイル表）で示すようなデータをDISK315に保持している。カラープロファイルマネージャ416で管理されるカラープロファイルは、各プリンタコントローラ（制御プログラム）510、511、512およびスキャナコントローラ（制御プログラム）420に渡され、プリントジョブ・コピージョブおよびスキャンジョブの処理の際に使用される。417はPDL（ページ記述言語）の一種であるPS（登録商標）でかかれたドキュメントデータを処理し、画像データを出力するPDLラスタライザである。418はPDL（ページ記述言語）の一種であるPCL（登録商標）でかかれたドキュメントデータを処理し、画像データを出力するPDLラスタライザである。419はスキャンジョブマネージャであり、スキャンジョブの管理を行う。スキャンジョブマネージャ419は図22（属性表）で示されるようなデータをDISK315に保持している。スキャンジョブマネージャ419はスキャナコントローラ420にスキャンの実行を指示する。420はスキャナエンジン310を制御するためのスキャナコントローラである。スキャナコントローラ420はスキャナエンジン310の機能・状態および性能を表す、図44（属性表）で示される様なデータをDISK315に保持している。スキャンジョブマネージャ419およびスキャナコントローラ420はそれぞれ機能・性能を示すデータ（図22と図44）を有しているが、これらのデータで示される機能は必ずしも同一である必要はない。スキャンジョブマネージャ419が保持するデータ（図22）をSupervisor410を通して書き換えることにより、機能を制限することが可能である。

【００３２】上述したフォントマネージャー413、フォームオーバーレイマネージャー414、ログマネージャー415、カラープロファイルマネージャー416は、リソース管理プログラムであり、スキャンジョブマネージャー419、プリントジョブマネージャー501から509、コピージョブマネージャー601から608が論理デバイス（スキャナ、プリンタ）制御プログラムであり、レーザービームプリンタコントローラー510,511、インクジェットプリンタコントローラー512、スキャナコントローラー420が

物理デバイス（プリンタ、スキャナ）制御プログラムである。

【００３３】（プリントジョブマネージャの構成）図5は、図4で示したソフトウエア（制御プログラム）構成図を補足する図である。ディスパッチャー412は図４で示すディスパッチャー412と同じものを表わし、プリントジョブマネージャ（制御プログラム）501から509と513へコマンドパケットの配布を行う。501から509までと513は、プリントジョブマネージャでありプリントジョブの管理を行う。各プリントジョブマネージャは図18（属性表）で示す様なデータをDISK315にそれぞれ有している。各プリントジョブマネージャは、プリントジョブを処理するために使用するプリンタコントローラ（LBP Controller510或いはLBP Controller511或いはInk Jet Controller512或いはController510、511、512の任意な組み合わせ）およびそれに接続されたプリントエンジンが決められており、どのプリンタコントローラ（制御プログラム）を使用するかは図18の属性ID2001で示されるデータに記述されている。ただしプリントジョブマネージャ508と513だけは、動的に使用するプリントエンジンを選択する機能を有しており、この機能を有しているということがプリントジョブマネージャ508と513の有するデータ（属性表）に記述されている。各プリントマネージャは入力されたプリントジョブ、およびこれらがどのプリンタコントローラでどのようなジョブとして実行されているかを表わす、図20（ジョブ表）、図21（ジョブ依頼表）、図90（ジョブスケジューリング表）、図91（実行時刻順ジョブキュー表）に示すようなデータをDISK 315に保持している。510から512までは、各プリントエンジン308、311および312を制御するプリンタコントローラである。プリンタコントローラ510はフィニッシャー309も制御する。各プリンタコントローラは対応するプリンタエンジンの機能・状態および性能を表す、図40（属性表）で示す様なデータをDISK315に有している。各コントローラは入力されたプリントジョブの状態を表わす図42（ジョブキュー表）に示すようなデータをDISK315に保持している。

【００３４】（割り込みプリントジョブマネージャの機能）図５のプリントジョブマネージャ513は他のプリントジョブマネージャにない特別な機能を有しており「割り込みプリントジョブマネージャ」と呼ばれている。すなわち、現在各プリントジョブマネージャが各プリントコントローラに対して投入・実行中のプリントジョブの少なくとも一つに割り込み、先にジョブの実行を行わせることができる機能を有している。割り込まれた実行中のジョブはプリントエンジンの許す限りの区切りのよい時点（例えば、各ページの排紙時点）で実行を中断することになる。本実施例においてはプリントエンジンはジョブの概念を持たず単に1枚単位の記録用紙上に画像を記録して排出するという単機能を繰り返して行っているにすぎない。

【００３５】従って、実質的なプリントジョブの割り込みは、プリントコントローラの機能によって実現されている。割り込みプリントジョブマネージャ513は割り込みというジョブの緊急性を考慮してシステムに存在する全プリントコントローラを動的に選択してジョブを行わせる機能を有している。実際には各プリントコントローラのうちからジョブを実行中でないものかエンジンのプリントスピードの速いものを有するプリントコントローラを一つを選択し、そのプリントコントローラの持つ割り込み制御機能に対して割り込みの指示と共にプリントジョブを発行する。割り込みプリントジョブマネージャ513も他のプリントマネジャと同様に図90（ジョブスケジューリング表）、図91（実行時刻順ジョブキュー表）に示すようなデータをDISK 315に保持しているが、その機能の性格上、プリントジョブマネージャ513に対するジョブ実行優先度やジョブ実行時刻指定は意味がない。従って、割り込みプリントジョブマネージャ513に対して上記オプション（ジョブ実行優先度、ジョブ実行時刻）を指定しても無視される。

【００３６】（プリントコントローラの機能）全てのプリントコントローラ（510,511,512）は、現在実行中のジョブの状況（コンテキスト）を変数として内部に保持している。例えばフィニッシャユニットを備えたプリントエンジン308の出力ビンの位置（ビンID）とか、自身の保持するジョブファイルの読み出し位置（ファイルポインタ）や残りプリント枚数などが相当する。また、全てのプリントコントローラは、これらのコンテキストをストアする領域をDISK315内の通常の保持領域とは別に備えている。さらに、ジョブ実行中もジョブ割り込み指示を監視しており、この指示と共にジョブが投入された場合には、現在実行中のジョブのコンテキストを前記ストア領域に待避したうえで投入された割り込みジョブ用に初期化する。その後、プリントエンジンに対して画像の記録を行わせていく。割り込みジョブの実行中は新たな割り込み指示は受け付けられない。割り込みジョブが終了した時点で、待避してあった被割り込みジョブのコンテキストが復帰され割り込まれた時点のジョブを再開する。

【００３７】プリントジョブマネージャ501から509と513およびプリンタコントローラ510から512はそれぞれ機能・性能を示す図18（属性表）や図40（属性表）のようなデータをDISK315に有している。一般的にプリントジョブマネージャは使用する各コントローラが有する性能の総和にあたる性能を有し、各コントローラが有する機能の共通項にあたる機能を有している。しかし、これらの一部の機能はプリントジョブマネージャが保持するデータ（図18）をSupervisor410を通して書き換えることにより、変更することが可能である。たとえば、プリントジョブマネージャ503とプリントジョブマネージャ509

は、同じプリンタコントローラ512を使用しているために、同じ性能・機能を提供することができるが、プリントジョブマネージャの保持するデータの後述する設定処理により、プリントジョブマネージャ503はカラー印刷可能でありプリントジョブマネージャ509はカラー印刷不可能とすることも出来る。また、割り込みプリントジョブマネージャ513以外の各プリントジョブマネージャは、属性ID2006でジョブ管理設定（ジョブ実行優先度、ジョブ実行時刻、ジョブ残存限度時刻）が可能かどうかが設定される。この属性は、デフォルトでYESに設定されている。

【００３８】（コピージョブマネージャーの構成）図６は、図4および図５で示したソフトウエア（制御プログラム）構成図を補足する図である。ディスパッチャー412は図４で示されているディスパッチャー412と同じものを表わし、コピージョブマネージャ（制御プログラム）601から６０９へコマンドパケットの配布を行う。６０１から609までは、コピージョブマネージャでありコピージョブの管理を行う。各コピージョブマネージャは図24（属性表）で示される様なデータをそれぞれDISK315に有している。各コピージョブマネージャは、コピージョブを処理するために使用するプリンタコントローラ、スキャナコントローラおよびそれに接続されたプリントエンジン、スキャナエンジンが決められており、どのプリンタコントローラとスキャナコントローラを使用するかは図24で示されるデータ（属性ID2001）に記述されている。ただしコピージョブマネージャ608と609だけは、動的に使用するプリントエンジンを選択する機能を有しており、この機能を有しているということがコピージョブマネージャ608と609の有するデータ（属性ID2002）に記述されている。図24につていは、詳細に後述するが、前述した図18,図19と同様に、割り込みコピージョブマネージャ609以外の各コピージョブマネージャは、属性ID2006でジョブ管理設定（ジョブ実行優先度、ジョブ実行時刻、ジョブ残存限度時刻）が可能かどうかが設定される。この属性は、デフォルトでYESに設定されている。

【００３９】各コピーマネージャは入力されたコピージョブ、およびこれらがどのプリンタコントローラ、スキャナコントローラでどのようなジョブとして実行されているかを表わす、図20（ジョブ表）や図21（ジョブ依頼表）、図90（ジョブスケジューリング表）、図91（実行時刻順ジョブキュー表）に示すようなデータをDISK315に保持している。510から512までは、各プリントエンジンを制御するプリンタコントローラであり、図５に示されているものと同一のものを表わしている。コピージョブマネジャはスキャナコントローラを用いて原稿を読み取らせてイメージデータを受け取る点が、プリントジョブマネジャとは異なるが、その後のプリントコントローラに対してジョブを投入する手順はまったく同一である。

【００４０】（割り込みコピージョブマネージャの機能）図６のコピージョブマネージャ609は他のコピージョブマネージャにない特別な機能を有しており「割り込みコピージョブマネージャ」と呼ばれている。すなわち、現在各コピージョブマネージャが各プリントコントローラに対して投入・実行中のプリントジョブの少なくとも一つに割り込み、先にジョブの実行を行わせることができる機能を有している。コピージョブマネジャはスキャンコントローラに対してジョブを投入した後にプリントコントローラに対してジョブを投入するという2段階でコピージョブを実行する。本実施例においては割り込みコピージョブは前記スキャンコントローラに対して投入するジョブには割り込むことは行わず、プリントコントローラに投入中のジョブにのみ割り込むようになっている。すなわち、スキャンエンジンで読み込んだイメージデータを当初予定していたプリンタエンジンを使わず他のプリンタエンジンを使って割り込み処理を行わせる。

【００４１】（Supervisorの属性表）図7は、Supervisor（制御プログラム）410がDISK315に保持し管理するデータ（属性表）を表わす。この表は多機能周辺機器201の機能概要や接続情報およびセキュリティ情報などを表わすものである。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードは属性ID701、型ID702および属性値703から構成されており、Supervisorの持つ属性を表わす。属性ID701は情報の種類を表わすものであり、これによって値703が示しているものがどのような意味を持つのかが示されている。属性ID701は機器内部でユニークであり、同じ属性IDを持つものは、同じ情報の種類を表わしている。型ID702は値703がどのようなデータ型を持っているのかを示すものであり、値703の解釈を行う際に使用されるものである。型IDは属性IDによって一意に決められるものであり機器内部でユニークに定義されている。この実施例においては属性表の中に属性ID701と型ID702の両方が入れられているが、属性IDと型IDとの対応表を属性表とは別のデータとして保持し、属性表の中には属性IDと値のみを入れる様に実施してもよい。値703は、属性ID701に従った属性値を示す。

【００４２】図8は、図7の属性表の属性ID701と型ID702の詳細を示す図である。この中で、属性ID102の「サポートしているセキュリティレベルのリスト」はSupervisorに対する操作を発行するときに必要とされるセキュリティレベルについて、設定可能なセキュリティレベルのリストを保持している。セキュリティレベルについては図14のアクセス制御表の説明の中でなされる。属性ID103 の「現在のセキュリティレベル」は、現在Supervisorに設定されているセキュリティレベルを表わす。属性ID401の「現在のカウントデータ」はSupervisorの中でカ

ウント対象となっている情報のリストを表わしており、この意味するところは属性ID402の「カウントデータフォーマット」で表わされている。例えば、カウントデータ45,78,34,13はカウントデータフォーマットで表わされた紙サイズのプリント枚数を示す。属性ID402の「カウントデータフォーマット」は、属性IDのリストとして表わされる。属性IDは機器内でユニークに定義されているため、属性IDを指定することによりカウントデータの意味を指定することが可能となっている。例えば、属性ID402の属性値565が用紙サイズA2のプリント枚数、属性値537が用紙サイズA3のプリント枚数、属性値545が用紙サイズA4のプリント枚数、属性値523が用紙サイズA5のプリント枚数を意味する。属性ID403のカウント制限データは、属性ID401の「現在のカウントデータ」が持つことが出来る最大値を表わしており、「現在のカウントデータ」の値のうち１つでもこの最大値を超えた場合、CPU301が図9で示されるサブアドレス：サービスID対応表の有効フラグを無効にすることにより全ての（マネージャによる）サービスを無効とする。属性ID404のカウント単価データは、属性ID402の「カウントデータフォーマット」に保持されている属性IDの1カウント毎の単価を通貨単位で表わすものである。属性ID405の現在の課金データは、属性ID401の「現在のカウントデータ」と属性ID404の「カウント単価データ」の対応する値を掛け合わせて総和を取った値を示すものである。属性ID406の課金制限データは、属性ID405の「現在の課金データ」の持つことが出来る最大値を表わしており、属性ID405の「現在の課金データ」が属性ID406の「課金制限データ」を超えた場合、CPU301が図9で示されるサブアドレス：サービスID対応表の有効フラグを無効フラグにすることにより全ての（マネージャによる）サービスを無効とする。属性ID501のサポートしているイベントリストは、Supervisorに対して通知することを設定可能なイベントを、イベントの種類毎に一意に定義されているイベントIDのリストによって表わしている。実際のイベント送信の設定は、図16で示されるイベント設定表にイベント通知の接続形式および宛先をイベントIDと共に設定することによって行う。図7の属性表の値について、「Table Data」と記述してあるものは、値が型IDで示されている表形式（52から104）を持つデータであることを示す。

【００４３】（サブアドレス：サービスID対応表）図9はサブアドレス：サービスID対応表を表すものである。この表は図7で示される属性表の属性ID1001（サブアドレス：SID対応表）の属性値としてDISK315に保持されている。サブアドレス：SID対応表はどのサブアドレスにコマンドパケットを投入すればどのサービス（例えばプリント、フォントのダウンロードなど）が得られるを示すものである。また、この表の有効フラグ等に従ってインタープリター409は各接続形態ごとにジョブサービスの提供をおこなうかどうかを制御する。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードは接続形式ID801、サブアドレス802、サービスＩＤ803、有効フラグ804、有効ユーザのリスト805および無効ユーザのリスト806から構成されている。接続形式ID801は接続形態を表す識別子を示している。図10は接続形式IDの意味を示す図であり、0はInternal（ユーザインターフェース）、1はTCP/IP（ネットワーク）、2はIEEE1284.4（IEEE1284）、3はSBPー2(IEEE1394)を表している。

【００４４】サブアドレス802は各接続形態ごとのサブアドレスを表している。ユーザインターフェースはサブアドレスを有しないが、便宜上サブアドレスを割り当てており、このサブアドレス情報とともにユーザインターフェースからコマンドパケットを投入する。

【００４５】サービスID803はサービスを表す識別子である。サービスID803は図4、図5および図6で示された各マネージャに一対一で割り当てられている。

【００４６】有効フラグ804は、そのサブアドレスが有効かどうかを表している。この値が真（YHS）の場合にはサブアドレスが有効であり、ユーザーがコマンドパケットをサブアドレスに対して発行することが出来る。値が偽（NO）の場合にはサブアドレスは無効であり、コマンドパケットをサブアドレスに対して発行してもパケットは破棄されることを意味する。図8の場合は、接続形式ID2（IEEE1284）の接続形態を利用した場合サービスID8（プリントジョブマネージャー）のサービスを提供できない事を表している。

【００４７】有効ユーザのリスト805は、サブアドレスが有効の場合に、そのサービスが受けられるユーザIDのリストを表している。無効ユーザのリスト806は、サブアドレスが有効の場合でも、そのサービスが受けられないユーザIDのリストを表している。有効ユーザのリスト805と無効ユーザのリスト806は、どちらかにしか値を設定することは出来ない。

【００４８】（サービスID：タスクタイプ対応表）図11はサービスID：タスクタイプ対応表を表すものである。この表は図7で示される属性表の属性ID1002（SID : Task Type表）の属性値としてDISK315に保持されている。サービスID：タスクタイプ表はサービスIDがどのような種類のサービスを提供するのかを表すものである。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードはサービスID901およびタスクタイプ902から構成されている。タスクタイプ902はどのような種類のサービスであるかを示している。図12はタスクタイプ902の値の意味を示す図である。0はSupervisorが行うサービス、1はプリント、２はスキャン、3はコピー、11は割り込みプリント、13は割り込みコピー、101はフォント、102はフォームオーバーレイ、103はログ、104はカ

ラープロファイルを表している。サービスIDはサービスを提供するマネージャと一対一で対応している。このため、サービスIDは各マネージャが有する機能表のアクセスにも使用される。機能表は各プリンタコントローラおよびスキャナコントローラも有しているために、これらにアクセスするためにプリンタコントローラおよびスキャナコントローラにもサービスIDに相当するコントローラIDを割り当てており、コントローラIDがどのような種類のコントローラであるかをサービスID：タスクタイプ対応表で管理している。タスクタイプ201はプリンタコントローラ、202はスキャナコントローラを表している。コントローラIDは図４および図５で示された各コントローラと一対一で対応している。

【００４９】サービスIDの意味は以下の通りである。
0 : Supervisor410
1 ： プリントジョブマネージャ501
2 ： プリントジョブマネージャ502
3 ： プリントジョブマネージャ503
4 ： プリントジョブマネージャ504
5 ： プリントジョブマネージャ505
6 ： プリントジョブマネージャ506
7 ： プリントジョブマネージャ507
8 ： プリントジョブマネージャ508
9 ： プリントジョブマネージャ509
10 ： スキャンジョブマネージャ419
11 ： コピージョブマネージャ601
12 ： コピージョブマネージャ602
13 ： コピージョブマネージャ603
14 ： コピージョブマネージャ604
15 ： コピージョブマネージャ605
16 ： コピージョブマネージャ606
17 ： コピージョブマネージャ607
18 ： コピージョブマネージャ608
19 ： 割り込みプリントジョブマネージャ513
20 ： 割り込みコピージョブマネージャ609
101 ： フォントマネージャ413
102 ： フォームオーバーレイマネージャ414
103 ： ログマネージャ415
104 ： カラープロファイルマネージャ416
21 ： プリンターコントローラ510
22 ： プリンターコントローラ511
23 ： プリンターコントローラ512
24 ： スキャナーコントローラ420

【００５０】（ユーザー認証表）図13はSupervisorが保持するユーザー認証表を表すものであり、機器を使用出来るユーザーの認証情報を表わす。この表は図7で示される属性表の属性ID1003（ユーザー認証表）の属性値としてDISK315に保持されている。ユーザー認証表は有効なユーザIDとパスワードの組を、そのユーザが管理者特権を有するかどうかの情報と共に表わすものである。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードはユーザID1001、パスワード1002および管理者特権フラグ1003から構成されている。管理者特権フラグ1003は、ユーザが管理者としての権利を有するかどうかを表している。

【００５１】（アクセス制御表）図14はSupervisorが保持するアクセス制御表を表すものであり、サービス毎のセキュリティレベルを表わす。この表は図7で示される属性表の属性ID1004（アクセス制御表）の属性値としてDISK315に保持されている。アクセス制御表は、サービスIDごとに必要なセキュリティレベル、およびサービス提供を許されているユーザのリストを保持している。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。

【００５２】各レコードはサービスID1101、セキュリティレベル1102およびユーザIDのリスト1103から構成されている。セキュリティレベル1102はサービスID1101で指定されたサービスを受ける時、つまりサービスID1101で示されるマネージャにコマンドパケットを発行する際に必要な認証情報を表わしている。図15はセキュリティレベルの意味を示す図である。セキュリティレベル0はユーザ認証を必要としないということ、1は管理者のみ区別をするということ、2は認証されたユーザのみ使用可能であるが、認証にユーザーIDを使用しパスワードは使用しないということ、3は認証されたユーザのみ使用可能であり、認証にユーザーIDとパスワードを使用することを表す。ユーザIDのリスト1103は、セキュリティレベル1102が2と3の場合に、使用（アクセス）を許可されたユーザIDのリストを表す。

【００５３】（イベント設定表）図16はSupervisorが保持するイベント設定表を表わすものである。この表は図7で示される属性表の属性ID502（イベント設定表）の属性値としてDISK315に保持されている。イベント設定表は指定されたイベントが機器内で起きた時にイベント通知を送信する方法と宛先を、イベントの種類毎に保持している。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードはイベントID1201、接続形式1202および通知先アドレス1203から構成されている。宛先は、接続形式と接続形式に依存した通知先アドレスとから構成される。イベントID1201は、イベントの種類を表わすものであり、機器内でユニークに定義されている。例えば、Event ID200は紙なし、ID399はトナーなし、ID432はインクなし、ID234は多機能周辺機器のカバーオープンを示す。接続形式ID1202は、イベント通知を送信する接続形式を表わしており、この値は図9のサブアドレス：サービスID対応表で使用されているものと同一のものが使用される。通知先アドレス1203は、接続形式1202に適合したイベントの通知先宛先を表わすものである。

【００５４】（イベントフォーマット表）図17はSupervisorが保持するイベントフォーマット表を表わすものでありイベントの内容を示す。この表は図7で示される属性表の属性ID503（イベントフォーマット表）の属性値としてDISK315に保持されている。イベントフォーマット表は機器内でユニークに定義されている各イベントIDについて、イベント通知として送信される付加データの形式を保持している。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードはイベントID1301およびイベントフォーマット1302から構成されている。イベントフォーマット1302はイベント通知として送信される付加データの形式であり、属性IDのリストとして表わされる。属性IDは機器内でユニークに定義されており、その型も属性IDによって一意に決定されているため、属性IDを指定することにより、付加データのフォーマットを表わすことができる。例えば、イベントフォーマット1302の属性ID676は紙サイズ、ID756は紙の種類、ID666はトナーの種類、ID698はインクの種類、ID600はカバーの位置を示す。以下に示す様に、イベントの送信方法・宛先は各マネージャおよびコントローラが保持するデータ（属性表）に設定されており、これらの各マネージャおよびコントローラ内で起きたイベントを通知することが可能となっているが、これらのイベントの付加データのフォーマットも図13で示されるイベントフォーマット表において保持されている。指定されたイベントが起きた時、イベントフォーマットで定義されている内容が、イベントID毎に予め決められたデータと共に通知される。

【００５５】（プリントジョブマネージャーの属性表）図18は各プリントジョブマネージャ（制御プログラム）501から509と513がDISK315に保持するデータ（属性表）を表わす。この表は、プリントジョブマネージャが扱うことが可能なプリントジョブの性能・機能を表わすものである。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードの内容は図7で示されるSupervisorのものと同一であり、属性ID1401、型ID1402および値1403から構成されている。図19は図18の属性表の属性ID1401と型ID1402の値を示す図である。図19の属性表の型ID203（ジョブ表形式）、型ID204（ジョブ依頼表形式）、型ID205（ジョブスケジューリング表形式）、型ID81（禁止属性組み合わせリスト1形式）、型ID82（禁止属性組み合わせリスト2形式）、型ID83（禁止属性組み合わせリスト3形式）、型ID84（禁止属性組み合わせリスト4形式）、型ID85（禁止属性組み合わせリスト5形式）以外の他の型IDについては図7のSupervisorの属性表で使用されているものと同一である。

【００５６】属性ID601の「サポートしているデータのダウンロード方法」は、プリントするドキュメントデータの機器への送信方法を表わすものであり、本実施形態では1：ジョブの中にドキュメントデータを含める方法、および2：ジョブの中はドキュメントデータの参照ポインタ（URL: Unified Resource Locator)を含めておき、機器から必要に応じて参照ポインタで示されるドキュメントデータを読み出す方法がサポートされている。

【００５７】属性ID801、802、803、804、805はプリントジョブマネージャに対して投入されるプリントジョブの中に設定する属性に関する制限を表す禁止設定属性である。これらは、例えばコピー部数が100を超える場合は、フィニッシャの設定が不能になるなどといった制限事項を表すために使用される。禁止設定属性には、その制限の表現方法として５つの型がある。属性ID801は、｛属性ID1：｛属性ID2｝｝で示す様に、ある属性ID1に対して別の属性ID2のリストの対を複数保持している。これは属性ID1の設定がなされている場合は各属性ID2の設定をプリントジョブスクリプトの中で行えないことを意味している。

【００５８】属性ID802は、｛属性ID1：演算子識別子：値：｛属性ID2｝｝で示す様に、ある属性ID1、演算子識別子および値に対して、別の属性ID2のリストの対を複数保持している。これは属性ID1に対して演算子識別子で示される演算子を値に対して適用した結果が真の場合は、各属性ID2の設定をプリントジョブスクリプトの中で行えないことを意味している。演算子識別子は整数値が2項演算に対して一対一で対応している。例えば、0は「＝（等しい）」を表す。

【００５９】属性ID803は、｛属性ID1：演算子識別子1：値1：｛属性ID2：演算子識別子2：値2｝｝で示す様に、ある属性ID1、演算子識別子1および値1に対して，別の属性ID2，演算識別子2および値2の組のリストの対を複数保持している。これは属性ID1に対して演算子識別子1で示される演算子を値1に対して適用した結果が真の場合は、各属性ID2に対して演算子識別子2示される演算子を値2に対して適用した結果が真になる様な設定はプリントジョブスクリプトの中で行えないことを表す。

【００６０】属性ID804は、｛(属性ID1：演算子識別子1：値1)：演算識別子3：（属性ID2：演算子識別子2：値2）：｛属性ID3｝｝で示す様に、ある属性ID1、演算子識別子1および値1と演算識別子3と別の属性ID2、演算識別子2および値2とに対して，さらに別の属性ID3のリストの対を複数保持している。これは属性ID1に対して演算子識別子1で示される演算子を値1に対して適用した結果と属性ID2に対して演算子識別子2で示される演算子を値2に対して適用した結果とを演算子識別子3で示される演算をした結果が真の場合は、各属性ID3に対する設定はプリントジョブスクリプトの中で行えないことを表す。

【００６１】属性ID805は、｛(属性ID1：演算子識別子1：値1)：演算識別子3：（属性ID2：演算子識別子2：値

2）：｛属性ID4：演算識別子4：値4｝｝で示すに、ある属性ID1、演算子識別子1および値1と演算識別子3と別の属性ID2、演算識別子2および値2とに対して，さらに別の属性ID4、演算識別子4および値4の組のリストの対を複数保持している。これは属性ID1に対して演算子識別子1で示される演算子を値1に対して適用した結果と属性ID2に対して演算子識別子2で示される演算子を値2に対して適用した結果とを演算子識別子3で示される演算をした結果が真の場合は、各属性ID4に対して演算し4で示される演算を値4に対して適用した結果が新になる様な設定はプリントジョブスクリプトの中で行えないことを表す。

【００６２】属性ID2001の「Jobを実行する（可能性のある）コントローラ IDリスト」は、プリントジョブを実行するコントローラのコントローラIDのリストであり、プリントジョブマネージャ508以外のプリントジョブマネージャは必ずジョブを実行するコントローラのリストとなっている。プリントジョブマネージャ508はコントローラをジョブの要求している機能によって動的にコントローラを選択するため、この属性はコントローラを選択する範囲を表わすものとなっている。属性ID2002の「コントローラ自動選択かどうか」は、プリントジョブマネージャがコントローラを自動選択するかどうかが表わされている。本実施形態では、プリントジョブマネージャ508のみがこの値を真としている。他の属性IDについては図7のSupervisorの属性表で使用されているものと同一である。ただし、設定の範囲および影響が及ぶ範囲は、属性表が属しているプリントジョブマネージャの管理下にあるものに限定されている。例えば、属性値101の「サポートしているオペレーション」は、このプリントジョブマネージャに対して発行できるオペレーションのリストであり、属性値406の「課金制限データ」はこのプリントジョブマネージャで実行したプリントジョブに関して課金の制限を加えるものである。課金の制限を超えた場合の取り扱いも、このプリントジョブマネージャのサービスを無効とするのみで、他のマネージャのサービスには影響を与えない。なお、図18に示すプリントジョブマネージャーの属性表はサービスIDが4の属性表（プリントジョブマネージャー504）であり、サービスID1,2,3,5,6,7,8,9,13のそれぞれに対応したプリントジョブマネージャー501,502,503,505,506,507,508,509,513の属性表がDISK315に記憶されておりそれぞれ異なる。

【００６３】（ジョブ表）図20は各プリントジョブマネージャが保持するデータ（ジョブ表）を示す図であり、プリントマネージャーが管理するジョブの実体を保持するファイル名とジョブIDとの対応表を示す。表は図18で示される属性表の属性ID2003（ジョブ表）の属性値としてRAM302に保持されている。各プリントジョブマネージャーがRAM302上のジョブ表を動的に変更し必要に応じてDISK315に保存する。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードはジョブID1501およびジョブの実体が保持されているファイル名1502、実行優先度1506、指定実行時刻1507、残存限度時刻1508から構成されている。ジョブID1501は、Client PC（202から205）からジョブパケットが投入され、プリントジョブマネージャにジョブが投入されたときに、プリントジョブマネージャが割り当てたジョブに対する識別子である。ジョブのファイル名1502は、ジョブの実体が保持されているファイルの名前である。ジョブの実体は図52に示す様に、属性ID、属性値サイズおよび属性値の組を複数持つことにより構成されている。

【００６４】詳細は後述するが、実行優先度1506、指定実行時刻1507、残存限度時刻1508は、プリントジョブマネージャにジョブが投入された時点で同時に、Client PC（202から205）からのジョブパケットに含まれる指定情報に基づいて指定される属性であり、全てその指定値が図20に示すジョブ表に格納される。実行優先度1506は範囲を持った自然数をその内容として持つ。本実施例では優先度は１から３の範囲をとり、この値（優先度）は小さいほどジョブの実行優先度が高いことを意味する。ジョブの依頼者がそのジョブの実行を比較的早く行わせたい場合にこの値を２ないし１に指定することができる。もし指定がなければデフォルトの優先度３が自動的に付与され本ジョブ表に格納される。ジョブは実行優先度に応じて後述するジョブスケジューリング表に登録され優先度の高い順に関係するコントローラに送られる。なお、この優先度はジョブを投入した後にSupervisor経由で変更することが可能である。その際にはジョブ表の優先度の値が変更されると共に、直ちにスケジューリング表が更新され新たな優先度に従ったジョブの実行スケジューリングが行われる。

【００６５】指定実行時刻1507はジョブの実行時刻を指定する必要がある場合に指定する。指定がなければジョブの実行は前述の優先度に基づいて処理される。指定実行時刻が指定されている場合には実行優先度の指定によらず優先度は強制的に1（最低優先度）に設定される。指定実行時刻はジョブが投入された時刻より未来に設定されることが必要である。指定実行時刻を指定されたジョブはジョブ投入時点ではジョブスケジューリング表にはすぐには登録されない。代わりに実行時刻順ジョブキュー表に登録される。指定実行時刻に達した時点で実行時刻順ジョブキュー表から登録を削除され、直ちに優先度１でジョブスケジューリング表に登録される。すなわち投入されたジョブは各ジョブマネージャのジョブスケジューリング表か実行時刻順ジョブキュー表のいずれか一方に必ず登録されることになる。なお、この指定実行時刻は、例えば、現在午前10時で、午前11時にジョブを実行したい場合には絶対時刻として午前11時をClient P

C（202から205）のユーザーインタフェースで指定できるし、今から30分以内にジョブを実行したい場合には相対時刻として今から30分以内をClient PC（202から205）のユーザーインタフェースから指定できる。前者は、例えば省エネルギーを考慮して夜間に大量のジョブを実行させたい場合や昼休み時間にジョブを実行させたい場合に有効であり、後者は現在時刻を考慮せずに、頭の中に描いている仕事のタイムスケジュールの中で自分の時間感覚でジョブを実行させたい場合に有効である。そして、相対時刻で指定実行時刻が指定された場合には、各ジョブマネージャにジョブが投入された時点でその時の現在の時刻と指定された相対時刻から絶対時刻をCPU301が計算して、図20のジョブ表、図84の実行時刻順ジョブキュー表に登録される。

【００６６】残存限度時刻1508属性はジョブの機器内における生存期限を定めるものである。通常のジョブは実行優先度、実行開始時刻、使用コントローラーの利用状況に応じて実行が遅延される場合がある。その際には機器内にしばらく未実行の状態でストアされたままとなっている。ところがジョブによってはその重要性、気密性の高さからあまりにも長期間機器内に未実行で放置したくないものがある。本実施例ではこのような要求に対応するためジョブの残存限度時刻を設定できるようになっている。すなわちこの属性が設定されたジョブはこの指定された時刻を超えて機器内に未実行状態で存在できず、自動的に削除される。残存限度時刻は先の実行開始時刻と併用して設定できる。残存限度時刻は実行開始時刻よりも未来の時刻を指定されることが望ましいが、もしその条件が満たされない場合にはジョブの投入直後にそのジョブは削除されることになる。なお、この残存限度時刻は、例えば、現在午前10時で、午前11時を限度としてジョブを削除したい場合には絶対時刻として午前11時をClient PC（202から205）のユーザーインタフェースで指定できるし、今から30分以内を限度としてジョブを削除したい場合には相対時刻として今から30分以内をClient PC（202から205）のユーザーインタフェースから指定できる。前者は、例えば、気密性の高いジョブを昼休み中に実行できない場合には昼休みの終わる時刻を限度時刻としてそのジョブを削除させたい場合に有効であり、後者は、例えば、頭の中に描いている仕事のタイムスケジュールの中で帰宅のため会社を出るために今から30分以内を限度時刻としてそのジョブを削除させたい場合に有効である。そして、相対時刻で残存限度時刻が指定された場合には、各ジョブマネージャにジョブが投入された時点でその時の現在の時刻と指定された相対時刻から絶対時刻をCPU301が計算して、図20のジョブ表、図84の実行時刻順ジョブキュー表に登録される。

【００６７】（ジョブ依頼表）図21はプリントジョブマネージャが保持するジョブ依頼表を表わす図でありプリントマネージャーが管理するジョブとコントローラーによって実行されるジョブとの関係を示す。この表は図18で示される属性表の属性ID2004（ジョブ依頼表）の属性値としてDISK315に保持されている。ジョブ依頼表は、プリントジョブマネージャが管理するジョブがどのコントローラでどのジョブとして実行されているかを表わすものである。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードはジョブID1503、コントローラID1504、およびコントローラの中で割り当てられているジョブID1505から構成されている。ジョブID1503はプリントジョブマネージャにジョブが投入されたときに、プリントジョブマネージャが割り当てたジョブに対する識別子であり、ジョブ表（図20）にあるジョブID1501と対応している。コントローラID1504はジョブが実行されているコントローラのIDを表わす。ジョブID1505は、ジョブを実行するコントローラが割り当てたジョブの識別子である。図21のジョブ表中のジョブID1503において同一の値を持ったレコードが複数存在しているがこれはある一つのジョブを複数のコントローラに分割して実行させる場合を示している。例えば、図21中のジョブID1503=1のジョブはコントローラID＝21、22の2つのコントローラへのジョブとして分割されそれぞれコントローラからジョブID1505＝100、101が割り当てられたことを示している。

【００６８】（ジョブスケジューリング表）図83は全プリントジョブマネージャ、全コピージョブマネージャが保持するジョブスケジューリング表を表す図である。ジョブスケジューリング表の各行は、ジョブ実行優先度毎におけるジョブIDのリストとなっている。すなわち、表の第一行目はジョブ実行優先度１のジョブIDが投入された時間順に左から記入されている（ジョブID5,7,8）。同様に2行目、3行目はそれぞれ実行優先度２、３のジョブ群が記入されている。ジョブスケジューリング表はジョブの投入、実行状況に応じて刻々変化してゆく。ジョブスケジューリング表内のジョブはジョブマネージャによって実行優先度の最も高くかつ投入順の最も若いものから取り出され（表から削除され）てコントローラに投入される。

【００６９】（実行時刻順ジョブキュー表）図84は全プリントジョブマネージャ、全コピージョブマネージャが保持する実行時刻順ジョブキュー表を表す図である。実行時刻順ジョブキュー表は、各行がジョブの実行されるべき絶対時刻とジョブIDからなる１レコードを成しており、表は複数のレコードの集合として構成されている。実行時刻1561とジョブID1562はそれぞれ、図21のジョブ依頼表の指定実行時刻1507とジョブID1503と同じ型である。図21のジョブ表の指定実行時刻1507に値が設定されているジョブはその指定実行時刻の若い順にソートされて実行時刻順ジョブキュー表にそのジョブIDと実行時刻が登録される。プリントジョブマネジャ、コピージョブ

マネージャに投入されたジョブのうち実行時刻指定がなされたものは一旦実行時刻順ジョブキュー表に登録されその後、実行時刻が到来したものが実行時刻順ジョブキュー表から削除されてジョブスケジューリング表に登録される。なお、この指定実行時刻は、前述したように、相対時刻或いは絶対時刻で指定されるが、相対時刻で指定実行時刻が指定された場合には、各ジョブマネージャにジョブが投入された時点でその時の現在の時刻と指定された相対時刻から絶対時刻をCPU301が計算して、図20のジョブ表、図84の実行時刻順ジョブキュー表に登録される。

【００７０】（スキャンジョブマネージャーの属性表）図22はスキャンジョブマネージャ419が保持するデータである属性表を表わす。この表は、スキャンジョブマネージャが扱うことが可能なスキャンジョブの性能・機能を表わすものである。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードの内容は図7で示されるSupervisorのものと同一であり、属性ID1601、型ID1602および値1603から構成されている。図23は図22の属性表の属性ID 1601と型ID1602の値を示す図である。属性ID=602 の「サポートしているデータのアップロード方法」は、スキャンしたドキュメントデータの機器への送信方法を表わすものであり、当実施例では1：ジョブの返信中にドキュメントデータを含める方法、および2：ドキュメントデータを機器内に保持しておきジョブの返信中にはドキュメントデータへの参照ポインタ（URL: Unified Resource Locator)を含めておき、ホスト（Client）から必要に応じて参照ポインタで示されるドキュメントデータを読み出す方法がサポートされている。属性ID=1201の「出力可能な画像フォーマットのリスト」は、出力可能な画像のデータフォーマットを表わしており、スキャンジョブを発行する際にこの中にあるものの1つをデータフォーマットとして指定することが出来る。属性ID＝2003のジョブ表に関しては、図20で示されたプリントジョブマネージャが保持するものと同一である。他の属性IDについては図7のSupervisorの属性表で使用されているものと同一である。ただし、設定の範囲および影響が及ぶ範囲は、属性表が属しているスキャンジョブマネージャの管理下にあるものに限定されている。図23の属性表の型ID203（ジョブ表形式）以外の他の型IDについては図7のSupervisorの属性表で使用されているものと同一である。

【００７１】（コピージョブマネージャーの属性表）図24は各コピージョブマネージャ601から609が保持するデータ（属性表）を表わす。この表は、コピージョブマネージャが扱うことが可能なコピージョブの性能・機能を表わすものである。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードの内容は図7で示されるSupervisorのものと同一であり、属性ID1701、型ID1702および値1703から構成されている。図25は図24の属性表の属性ID1701と型ID1702の値を示す。属性ID1302（カラー印刷可能か）、属性ID1303（サポートしているフィニッシングの種類）、属性ID1304（設定可能最高解像度）、属性ID1305（設定可能最低解像度）、属性ID2001（Jobを実行する可能性のあるコントローラ IDリスト）、属性ID2002（コントローラ自動選択かどうか）、属性ID2003（ジョブ表）、は、プリントジョブマネージャー及びスキャンジョブマネージャーの説明の中で示されたものと同一であり、属性ID2004（Job依頼表）、属性ID2005（Jobスケジュール表）、属性ID2006（ジョブ管理設定可能かどうか）は、プリントジョブマネージャの説明の中で示されたものと同一である。また、属性ID1302（カラー印刷可能か）、属性ID1303（サポートしているフィニッシングの種類）、属性ID1304（設定可能最高解像度）、属性ID1305（設定可能最低解像度）、属性ID2002（コントローラ自動選択かどうか）、属性ID2003（ジョブ表）、属性ID2004（ジョブ依頼表）、属性ID2005（ジョブスケジュール表）、属性ID2006（ジョブ管理設定可能かどうか）、型ID203（ジョブ表形式）、型ID204（ジョブ依頼表形式）、型ID205（ジョブスケジュール表形式）以外の属性IDと型IDについては図7のSupervisorの属性表で使用されているものと同一である。なお、図24に示すコピージョブマネージャーの属性表はサービスID が18の属性表（コピージョブマネージャー608）であり、サービスID11,12,13,14,15,16,17,19のそれぞれに対応したコピージョブマネージャー601,602,603,604,605,606,607,609の属性表がそれぞれDISK315に記憶されておりそれぞれ異なる。

【００７２】（フォントマネージャーの属性表）図26はフォントマネージャ413がDISK315に保持するデータ（属性表）を表わす。この表は、フォントマネージャが扱うことが可能なフォントのタイプや、現在フォントマネージャが管理しているフォントのリストなどを表わすものである。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードの内容は図7で示されるSupervisorのものと同一であり、属性ID1801、型ID1802および値1803から構成されている。図27は図26の属性表の属性ID1801と型ID1802の値を示す。属性ID601（サポートしているデータのダウンロード方法）、属性ID602（サポートしているデータのアップロード方法）、属性ID1501（サポートしているフォントタイプのリスト）、属性ID1502（最大保持可能フォント数）、属性ID1503（現在保持しているフォント数）、属性ID1504（保持しているフォントのリスト）、型ID150（フォント表）以外の属性IDと型IDは図7のSupervisorの属性表で使用されているものと同一である。ただし、設定の範囲および影響が及ぶ範囲は、属性表が属しているフォントマネージャの管理下

にあるものに限定されている。属性ID601の「サポートしているデータのダウンロード方法」、および属性ID602の「サポートしているデータのアップロード方法」はそれぞれプリントジョブマネージャの説明中およびスキャンジョブマネージャの説明中で説明されているものと同一であり、サポートしているフォントデータのダウンロード方法およびアップロード方法を示すものである。

【００７３】（フォント表）図28はフォントマネージャが保持するフォント表を表わすものである。この表は図26で示される属性表の型ID1504（フォント表）の属性値としてDISK315に保持されている。フォント表は、フォントマネージャがどの様なフォントを現在管理しているかを表わすものである。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードはフォントID1901、フォントタイプ1902、フォント名1903およびフォントデータを含むファイル名1904から構成されている。フォントデータはDISK315に記憶されており、フォントID1901はフォントデータがRAM302にダウンロードされたときに、フォントマネージャが割り当てたフォントに対する識別子である。

【００７４】（フォームオーバーレイマネージャーの属性表）図29はフォームオーバーレイマネージャ414がDISK315に保持するデータ（属性表）を表わす。この表は、フォームオーバーレイマネージャが扱うことが可能なフォームオーバーレイのフォーマットや、現在フォームオーバーレイマネージャが管理しているフォームオーバーレイのリストなどを表わすものである。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードの内容は図7で示されるSupervisorのものと同一であり、属性ID2001、型ID2002および値2003から構成されている。図30は図29の属性表の属性IDと型IDの値を示す。属性ID601（サポートしているデータのダウンロード方法）、属性ID602（サポートしているデータのアップロード方法）、属性ID1601（サポートしているフォームオーバーレイのフォーマットのリスト）、属性ID1602（最大保持可能フォームオーバーレイ数）、属性ID1603（現在保持しているフォームオーバーレイ数）、属性ID1604（保持しているフォームオーバーレイのリスト）、型ID160（フォーム表）以外の属性IDと型IDについては図7のSupervisorの属性表で使用されているものと同一である。ただし、設定の範囲および影響が及ぶ範囲は、属性表が属しているフォームオーバーレイマネージャの管理下にあるものに限定されている。属性ID601の「サポートしているデータのダウンロード方法」、および属性ID602の「サポートしているデータのアップロード方法」はそれぞれプリントジョブマネージャの説明中およびスキャンジョブマネージャの説明中で説明されているものと同一であり、サポートしているフォームオーバーレイデータのダウンロード方法およびアップロード方法を示すものである。

【００７５】（フォームオーバーレイ表）図31はフォームオーバーレイマネージャ414が保持するフォームオーバーレイ表を表わすものである。この表は図29で示される属性表の属性ID1604（フォームオーバーレイ表）の属性値としてDISK315に保持されている。フォームオーバーレイ表は、フォームオーバーレイマネージャがどの様なフォームオーバーレイを現在管理しているかを表わすものである。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードはフォームオーバーレイID2101、フォームオーバーレイのデータフォーマット2102、フォームオーバーレイ名2103およびフォームオーバーレイデータを含むファイル名2104から構成されている。フォームオーバーレイデータはDISK315に記憶されており、フォームオーバーレイID2101はフォームオーバーレイデータがRAM302にダウンロードされたときに、フォームオーバーレイマネージャが割り当てたフォントに対する識別子である。

【００７６】（ログマネージャーの属性表）図32はログマネージャ415がDISK315に保持するデータ（属性表）を表わす。この表は、現在ログマネージャが管理しているログのリストなどを表わすものである。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードの内容は図7で示されるSupervisorのものと同一であり、属性ID2201、型ID2202および値2203から構成されている。図33は図32の属性表の属性IDと型IDの値を示す。属性ID602（サポートしているデータのアップロード方法）、属性ID1703（保持されているLog数）、属性ID1704（ログ表）、属性ID1705（Logのフォーマット表）、型ID170（ログ表）、型ID171（ログフォーマット表形式）以外の属性IDと型IDについては図7のSupervisorの属性表で使用されているものと同一である。ただし、設定の範囲および影響が及ぶ範囲は、属性表が属しているログマネージャの管理下にあるものに限定されている。

【００７７】（ログ表）図34はログマネージャ415が保持するログ表を表わすものである。この表は図32で示される属性表の属性ID1704（ログ表）の属性値としてDISK315に保持されている。ログ表は、ログマネージャがどの様なログを現在管理しているかを表わすものである。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードはログID2301、およびログデータを含むファイル名2302から構成されている。ログID2301はあらかじめ種類に対して規定されているログの識別子である。

【００７８】（ログデータの内容）図35はログマネージャがDISK315で管理するログデータの内容を表わすもの

である。ログデータはログレコードの集合として構成される。各ログレコードはログフォーマットID2401、記録日時2402およびログデータ2403とから構成されている。ログフォーマットID2401は、図36で示されるログフォーマット表に記録されているログのフォーマット情報を参照しており、ログデータ2403のフォーマットを表わしている。例えば、LogFormatID1のログデータ「12345,4,1．23、“OK”」は、サービスID1のプリントジョブマネージャーにジョブを発行したユーザーIDが12345で、出力した紙数が4で、使用したトナー量が1．23、ジョブの終了状態がOKを示す。

【００７９】（ログフォーマット表）図36はログマネージャ415が保持するログフォーマット表を表わすものである。この表は図32で示される属性表の属性ID（ログフォーマット表）の属性値としてDISK315に保持されている。ログフォーマット表は、ログマネージャ415が管理する各ログデータ内で記録されているログのフォーマットを表わしている。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードはログフォーマットID2501、およびログフォーマット2502とから構成されている。ログフォーマットはサービスID：属性ID対のリストとして表わされる。属性IDは機器内でユニークに定義されており、その型も属性IDによって一意に決定されているため、属性IDを指定することにより、ログデータのフォーマットを表わすことができる。ログフォーマット2502のサービスID：属性ID対のリストと、ログデータ2403の中の値のリストは、順番によって対応している。例えば、ログフォーマット2502リストの3番目のサービスID：属性ID対はログデータの3番目の値のフォーマットを意味する。例えば、LogFormatID1のフォーマット「1：701,1：565,1：765,1：777」は、サービスID1のプリントジョブマネージャーにジョブを発行したユーザーID、出力した紙数、使用したトナー量、ジョブの終了状態を示す。また、LogFormatID2のフォーマット「11：701,11：565,11：765,11：777」は、サービスID11のプリントジョブマネージャーにジョブを発行したユーザーID、出力した紙数、使用したトナー量、ジョブの終了状態を示す。

【００８０】（カラープロファイルマネージャーの属性表）図37はカラープロファイルマネージャ416がDISK315に保持するデータ（属性表）を表わす。この表は、カラープロファイルマネージャが扱うことが可能なカラープロファイルのフォーマットや、現在カラープロファイルマネージャが管理しているカラープロファイルのリストなどを表わすものである。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードの内容は図7で示されるSupervisorのものと同一であり、属性ID2601、型ID2602および値2603から構成されている。図38は図37の属性表の属性IDと型IDの値を示す。属性ID601（サポートしているデータのダウンロード方法）、属性ID602（サポートしているデータのアップロード方法）、属性ID1801（サポートしているカラープロファイルデータのフォーマットのリスト）、属性ID1802（最大保持可能カラープロファイル数）、属性ID1803（現在保持しているカラープロファイル数）、属性ID1804（カラープロファイル表）、型ID180（カラープロファイル表形式）以外の他の属性IDについては図7のSupervisorの属性表で使用されているものと同一である。ただし、設定の範囲および影響が及ぶ範囲は、属性表が属しているカラープロファイルマネージャの管理下にあるものに限定されている。属性ID601の「サポートしているデータのダウンロード方法」、および属性ID602の「サポートしているデータのアップロード方法」はそれぞれプリントジョブマネージャの説明中およびスキャンジョブマネージャの説明中で説明されているものと同一であり、サポートしているフォントデータのダウンロード方法およびアップロード方法を示すものである。

【００８１】（カラープロファイル表）図39はカラープロファイルマネージャが保持するカラープロファイルを表わすものである。この表は図37で示される属性表の属性ID1804（カラープロファイル表）の属性値としてDISK315に保持されている。カラープロファイル表は、カラープロファイルマネージャがどの様なカラープロファイルを現在管理しているかを表わすものである。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードはカラープロファイルID2701、カラープロファイルフォーマット2702およびカラープロファイルデータを含むファイル名2703から構成されている。カラープロファイルＩＤ２７０１はカラープロファイルデータがダウンロードされたときに、カラープロファイルマネージャが割り当てたカラープロファイルに対する識別子である。カラープロファイルデータとは、Ｓｃａｎｎｅｒ Ｅｎｇｉｎｅ１０２から送られるデータや Ink Jet Printer Engine105へ送られるデータを補正してカラーマッチングを行うための補正データである。

【００８２】（プリンタコントローラの属性表）図40は各プリンタコントローラ510、511、512がDISK315に保持するデータ（属性表）を表わす。この表は、プリンタコントローラが制御しているプリントエンジンおよびフィニッシャーの性能・機能を表わすものであり、性能・機能に関する値は書き換えが出来ない。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードの内容は図7で示されるSupervisorのものと同一であり、属性ID2801、型ID2802および値2803から構成されている。図41は図40の属性表の属性IDと型IDの値を示す。属性ID12（コントローラタイプ）、属性ID13（コントローラI

D）、属性ID5001（カラー印刷可能かどうか）、属性ID5002（サポートしているフィニッシングの種類）、属性ID5003（設定可能最高解像度）、属性ID5004（設定可能最低解像度）、属性ID5005（ジョブキュー表）、型ID500（ジョブキュー表形式）以外の属性IDと型IDについては図7のSupervisorの属性表で使用されているものと同一である。ただし、設定の範囲および影響が及ぶ範囲は、属性表が属しているプリンタコントローラの管理下にあるものに限定されている。

【００８３】なお、図40に示すプリンタコントローラの属性表はサービスIDが21の属性表（プリンタコントローラ510）であり、サービスID22,23のそれぞれに対応したプリンタコントローラ511,512の属性表がDISK315に記憶されておりそれぞれ異なる。

【００８４】（ジョブキュー表）図42は全プリンタコントローラが保持するジョブキュー表を表わすものである。この表は図40で示される属性表の属性ID5005（ジョブキュー表）の属性値としてDISK315に保持されている。ジョブキュー表は、プリンタコントローラが管理・実行するジョブがどの様な状態にあるのかを表わすものである。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードはジョブID2901、ジョブステータス2902およびジョブの実体が保持されているファイル名2903から構成されている。ジョブID2901はプリンタコントローラにジョブが投入されたときに、プリンタコントローラが割り当てたジョブに対する識別子である。図43は図42のジョブステータス2902（ジョブの状態）を表わす図である。1はジョブの終了処理中、2はジョブがエンジンで実行中、3は実行待ち状態、4は割り込みジョブによる実行サスペンド（見合わせ）状態であることを示す。ジョブのファイル名2903は、ジョブの実体が保持されているファイルの名前である。ジョブの実体は図52に示す様に、属性ID、属性値サイズおよび属性値の組を複数持つことにより構成されている。

【００８５】（ジョブコンテキスト表）図85は全プリンタコントローラが保持するジョブコンテキスト表を表すものである。この表はプリンタコントローラが各プリントジョブマネジャ、コピージョブマネージャに対して発行したジョブID（1563）、投入されたジョブファイル内の位置を示すファイルポインタ（1564）、残コピー枚数（1565）、フィニッシャのビンID（1566）がその内容である。

【００８６】（スキャンコントローラーの属性表）図44はスキャナコントローラ420がDISK 315に保持するデータ（属性表）を表わす。この表は、スキャナコントローラが制御しているスキャナエンジン性能・機能を表わすものであり、性能・機能に関する値は書き換えが出来ない。表は各行が1つの情報単位（レコード）を表わしており、複数のレコードの集合としてデータは構成されている。各レコードの内容は図7で示されるSupervisorのものと同一であり、属性ID3001、型ID3002および値3003から構成されている。図45は図44の属性表の属性IDと型IDの値を示す。属性ID12（コントローラタイプ）、属性ID13（コントローラID）、属性ID6001（カラースキャン可能かどうか）、属性ID6002（最大原稿サイズ）、属性ID6003（設定可能最高解像度）、属性ID6004（設定可能最低解像度）以外の他の属性IDについては図7のSupervisorの属性表で使用されているものと同一である。ただし、設定の範囲および影響が及ぶ範囲は、属性表が属しているプリンタコントローラの管理下にあるものに限定されている。

【００８７】（コマンドパケットの構造）図46は、ユーザインターフェースマネージャ405、TCP／IP・UDP／IP処理モジュール406、IEEE1284.4処理モジュール407およびSBP−2処理モジュール408からインタープリンター409に対して出力されるコマンドパケットの構造を表わすものである。またこのコマンドパケットはインタープリタ409からユーザインターフェースマネージャ405、TCP／IP・UDP／IP処理モジュール406、IEEE1284.4処理モジュール407およびSBP−2処理モジュール408に出力される返信パケットおよびイベントパケットの構造も表わしている。パケットは、パケットの先頭を表わすパケットヘッダ3101、パケットの構造バージョンを表わすパケットバージョン3102、パケットの性格を表わすフラグ3103、どんな種類の操作を行うのかを表わすオペレーションコード3104、Client（PC）が返信パケットを認識するために使用するブロック番号3105、パラメータ3110の長さ表わすパラメータ長3106、ユーザの認証に使用されるユーザID3107とパスワード3108、返信パケットにのみ使用され返信の一般的な状態を表わすステータスコード3109およびオペレーションコード3105毎に決められたフォーマットを持つパラメータ3110から構成されている。パラメータ3110には、アクセス対象サービスID、アクセス対象属性ID等を含む。フラグ3104には、パケットがコマンドパケット・イベントパケットであるかまたは返信パケットであるかを表わすもの3111、およびパラメータ3110に送信すべきデータが入りきらず、次に送信されるパケットにも続きのデータが入っていることを示す連続フラグ3112とがある。パラメータ3110の長さは、パラメータ長3106が表わされる数（64kバイト）に制限されている。具体的には、Ethernet208に接続されているClientPC202,203はIPアドレス、ポート番号を指定して多機能周辺機器201に接続しIPパケットデータをNetwork Interface305(107)へ出力する。IEEE1394インタフェース206に接続されているClientPC204はノードIDとLUN（ロジカルユニットナンバー）を指定して多機能周辺機器201と接続しSBP−2パケットデータをIEEE1394 Interface306(108)へ出力する。IEEE1284インタフェース207に接続されているClientPC205はソケット番号を指定して多機能周辺機器20

1と接続しIEEE1284．4パケットデータをIEEE1284 Interface307(109)へ出力する。Network Interface Driver402，IEEE 1284 Interface Driver403，IEEE1394 Interface Driver404で図46に示すコマンドパケットの先頭にそれぞれIP Header，1284Header，1394 Headerが付加されたトランスポートパケットをTCP/IP・UDP/IP処理モジュール406，IEEE1284.4処理モジュール407、SBPー2処理モジュール408に出力する。TCP/IP・UDP/IP処理モジュール406,IEEE1284.4処理モジュール407、SBPー2処理モジュール408は入力した各トランスポートパケットの処理を行い図46に示すコマンドパケットを抽出しInterpreter／Generater409に出力する。

【００８８】（コマンドパケットの処理）図47は多機能周辺機器201におけるコマンドパケットの処理フローを表わすものである。ステップ3201において、各インターフェース401、402、403、404から入力されたデータを各トランスポート処理モジュール405、406、407、408が処理し図46に示すコマンドパケットを抽出する。抽出されたコマンドパケットは、データが入力された接続形式の情報（接続形式ID）およびサブアドレスと共にTCP/IP・UDP/IP処理モジュール406，IEEE1284.4処理モジュール407、SBPー2処理モジュール408からインタープリタ409に入力される。ステップ3202において、インタープリタ409はSupervisor410が保持するサブアドレス：サービスID対応表（図9）を参照し、入力された接続形式IDとサブアドレスとを比較することにより、サービスIDを得ると共にデータ入力が有効かどうかをチェックする。チェックの結果、データ入力が有効でない場合は、ステップ3204においてコマンドパケットを破棄して終了する。データ入力が有効の場合は、ステップ3203においてコマンドパケットの解析を図46のパケット構造に基づいて行う。パケットの解析の結果、図46で示される各項目は、それぞれ独立した別々の情報として出力される。ステップ3205において、サービスIDを基にアクセス制御表（図14）を参照することによりサービスIDに対応するセキュリティレベルを取得する。

【００８９】ステップ3206において、ステップ3205において取得したセキュリティレベルが0または1であるかを調べる。セキュリティレベルが0または1の場合は、セキュリティレベルのチェック（セキュリティゲート411）を行わずにディスパッチャ412に入力する。ステップ3206においてセキュリティレベルが0または1以外の時は、ステップ3207においてセキュリティレベルが2であるかどうかを調べる。セキュリティレベルが2の場合はステップ3209においてコマンドパケットに含まれていたユーザIDがユーザ認証表（図13）内に存在するかどうかをチェックする。ユーザIDが含まれていない場合、権限がないものとしてステップ3210においてエラー返信情報を生成し、インタープリンター409によってパケット（ステータスコード3109にエラー情報を埋め込む）を生成しコマンドパケット送信元に送信する。ユーザIDが含まれている場合、ディスパッチャ412に入力する。送信元アドレスはトランスポートパケット（Header）から抽出される。ステップ3207において、セキュリティレベルが2以外の場合は、ステップ3208においてコマンドパケットに含まれていたユーザID：パスワードの対がユーザ認証表（図13）内に存在するかどうかをチェックする。ユーザID：パスワードの対が含まれていない場合、権限がないものとして3211においてエラー返信情報を生成し、インタープリンター409によってパケットを生成しコマンドパケット送信元に送信する。ユーザIDが含まれている場合、ディスパッチャ412に入力する。ステップ3212においてディスパッチャ412は、サービスIDを元に配布先のマネージャを決定し、決定されたマネージャーに対してデータが入力された接続形式IDとサブアドレス、送信元のアドレス、オペレーションコード、ブロック番号、フラグ情報、ユーザＩＤ、パスワード、パラメータ長およびパラメータ（前述したジョブ管理設定情報であるジョブ実行優先度、ジョブ指定実行時刻、ジョブ残存限度時刻を含む）を配布する。そして、ステップ3213において各マネージャは、これらの情報を処理する。

【００９０】（属性表のアクセス処理）図48は、多機能周辺機器201が保持する各マネージャ・各コントローラの属性表に対する読み出し・書き込みを、Clientが指示した場合の処理を表す。各マネージャ・各コントローラが保持する属性表からの読み込みおよび書き込みは、Supervisorの有するサブアドレスに対して適当なコマンドパケットを送信することにより行う。属性表読み出し用コマンドパケットにはパラメータとして、アクセス対象サービスIDおよびアクセス対象属性IDが含まれている。また、属性表書き込み用コマンドパケットにはパラメータとして、アクセス対象サービスID、アクセス対象属性IDおよび属性IDに対応した属性値が含まれている。Clientから多機能周辺機器201に送られたパケットデータは、図47に示すフローによって処理され、Supervisorに配布される。ステップ3301においてオペレーションコード3104が、属性値読み出し用コード（”Get”）であるかどうかチェックする。オペレーションコードが”Get”である場合は、ステップ3301においてサービスIDを元にアクセス対象の属性表全体を取得する。サービスIDが0の場合は図7に示すSupervisor410の属性表、サービスIDが1,2,3,4,5,6,7,8,9の場合は図18に示すようなプリントジョブマネージャーの属性表、サービスIDが10の場合は図22示すようなスキャンジョブマネージャー419の属性表、サービスIDが11,12,13,14,15,16,17,18の場合は図24に示すようなコピージョブマネージャーの属性表、サービスIDが101の場合は図26に示すようなフォントマネージャー413の属性表、サービスIDが102の場合は図29に示すようなフォームオーバーレイマネージャー414の属性表、サービスIDが103の場合は図32に示すような

ログマネージャー415の属性表、サービスIDが104の場合は図37に示すようなカラープロファイルマネージャー416の属性表、サービスIDが21,22,23の場合は図40に示すようなプリンタコントローラの属性表、サービスIDが24の場合は図44に示すようなスキャナコントローラ420の属性表を取得する。なお、図18に示すプリントジョブマネージャーの属性表はサービスIDが4の属性表（プリントジョブマネージャー504）であり、サービスID1,2,3,5,6,7,8,9のそれぞれに対応したプリントジョブマネージャー501,502,503,505,506,507,508,509の属性表がDISK315に記憶されている。同様に、図24に示すコピージョブマネージャーの属性表はサービスID が18の属性表（コピージョブマネージャー608）であり、サービスID11,12,13,14,15,16,17のそれぞれに対応したコピージョブマネージャー601,602,603,604,605,606,607の属性表がそれぞれDISK315に記憶されている。また、同様に、図40に示すプリンタコントローラの属性表はサービスIDが21の属性表（プリンタコントローラ510）であり、サービスID22,23のそれぞれに対応したプリンタコントローラ511,512の属性表がDISK315に記憶されている。

【００９１】その後、ステップ3303において、指定された属性値が取得可能かどうかを検査する。検査は、対象となる属性表の「管理者のみが取得出来る属性ID リスト」（属性ID＝105）の値を取得し、この中に取得指示された属性IDが含まれているかどうかを調べることにより行う。属性IDが含まれていれば、値を取得することは出来ないので、ステップ3305においてエラー返信パケットを作成し、これを送信して終了する。属性IDが含まれていなければ、ステップ3304において指定された属性IDを元に属性表を検索し、該当する属性IDを持つ型IDと属性値を取得する。ステップ3306において、型IDに基づいて返信パケットを生成し、ステップ3307においてこの返信パケットを送信して終了する。ステップ3301においてオペレーションコードが”Get”でない場合は、ステップ3308において、オペレーションコードが、属性値書き込み用コード（”Set”）であるかどうかチェックする。オペレーションコードが”Set”である場合は、ステップ3309においてサービスIDに基づいてアクセス対象の属性表全体を取得する。その後、ステップ3310において、指定された属性値が設定可能かどうかを検査する。検査は、対象となる属性表の「管理者のみが設定出来る属性ID リスト」（属性ID＝104）の値を取得し、この中に設定指示された属性IDが含まれているかどうかを調べることにより行う。属性IDが含まれていれば、値を取得することは出来ないので、ステップ3314においてエラー返信パケットを作成し、これを送信して終了する。属性IDが含まれていなければ、ステップ3311において指定された属性IDを元に属性表を検索し、該当する属性IDを持つ型IDと属性値を取得する。ステップ3312において、型IDに基づいて指定された属性値を設定し、設定が成功した旨を知らせる返信パケットを生成する。ステップ3313においてこの返信パケットを送信して終了する。ステップ3308においてオペレーションコードが”Set”でない場合は、オペレーションコードに従った後述するその他の処理ステップ3315を行い終了する。

【００９２】（サービスID一覧問い合わせ処理）図49は、多機能周辺機器201が提供するサービスの一覧（サービスIDの一覧）をClientが問い合わせた場合の処理を表す。サービスの一覧の問い合わせは、Supervisorの有するサブアドレスに対して適当なコマンドパケットを送信して、図9で示されるサブアドレス：サービスID対応表を読み出し処理することにより行う。図49で示すフローは、図48で示したフローのステップ3315（その他のOperation Codeの処理）のステップを詳細化したものである。ステップ3401において、オペレーションコードがサービス一覧取得用コード（”List Service”）であるかどうかチェックする。オペレーションコードが”List Service” でない場合は、オペレーションコードに従った後述するその他の処理ステップ3402を行い終了する。ステップ3401において、オペレーションコードが “List Service”の場合は、ステップ3403において、サブアドレス：サービスID対応表の中から問い合わせに使用された接続形式IDに相当するものだけを抜き出したリストを作成する。ステップ3404において、ステップ3403で作成したリストの各レコードを検査し、有効フラグ804が有効、有効ユーザリスト805に値がある場合は問合わせに使用したユーザIDがその値に該当しないレコードを、無効ユーザリスト806に値がある場合は問い合わせに使用したユーザIDがその値に該当するレコードをリストから削除する。ステップ3405において、ステップ3404で更新したサービスIDのリストを含む返信パケットを作成し、作成された返信パケットは、ステップ3406において送信する。

【００９３】（サブアドレス問い合わせ処理）図50は、多機能周辺機器201が提供するサービスを利用するときに必要となるサブアドレスの情報を、Clientがサービス IDを指定して問い合わせた場合の処理を表す。サブアドレスの問い合わせは、Supervisor410の有するサブアドレスに対して適当なコマンドパケットを送信して、図9で示されるサブアドレス：サービスID対応表を読み出し処理することにより行う。サブアドレスの問い合わせコマンドパケットにはパラメータとして、サービスIDが含まれている。図50で示すフローは、図49で示したフローのステップ3402（その他のOperation Codeの処理）のステップを詳細化したものである。ステップ3501において、オペレーションコードがサブアドレス取得用コード（”Reserve”）であるかどうかチェックする。オペレーションコードが”Reserve” でない場合は、オペレーションコードに従った後述するその他の処理ステップ3503を行い終了する。ステップ3501においてオペレーショ

ンコードが“Reserve”の場合は、ステップ3502において問合わせに使用された接続形式IDとパラメータ内に指定されたサービスIDとからサブアドレス：サービスID対応表を走査し該当するレコードを検索する。ステップ3504において検索されたレコードについて、有効フラグが有効となっているか、有効ユーザリストに値がある場合はリスト中に問合わせに使用したユーザIDがあるかどうか、無効ユーザリストに値がある場合はリスト中に問い合わせに使用したユーザIDが無いかどうかをチェックする。チェックの結果どれかでもOKでない場合は、指定されたサービスに対するアクセス権（利用権）が無いものとして、ステップ3506においてエラー返信パケットを作成し、ステップ3507においてこのパケットを送信して終了する。ステップ3504におけるチェックの結果、どのチェックもOKの場合は、ステップ3505においてサブアドレスを含む返信パケットを作成し、ステップ3507においてこのパケットを送信して終了する。

【００９４】（タスクタイプ指定のサービスIDの問い合わせ処理）図51は、多機能周辺機器201が提供するサービスのうち、サービスの種類を指定して指定された種類のサービスを処理するために最適なサービスIDを問い合わせた場合の処理フローを表す。サービスの種類を指定したサービスIDの問合わせは、Supervisorの有するサブアドレスに対して適当なコマンドパケットを送信して、図9で示されるサブアドレス：サービスID対応表と図11で示されるサービスID：タスクタイプ対応表とを読み出し処理することにより行う。サブアドレスID問い合わせコマンドパケットにはパラメータとして、サービスの種類を指定するためのタスクタイプ、およびサービスを限定するための組み合わせ（条件情報）が含まれている。条件情報は属性IDと値の対のリストとして表されている。図51で示すフローは、図50で示したフローのステップ3503（その他のＯｐｅｒａｔｉｏｎ　Ｃｏｄｅの処理）のステップを詳細化したものである。ステップ３６０１において、オペレーションコードがサブアドレス取得用コード（”Get Service”）であるかどうかチェックする。オペレーションコードが”Get Service”でない場合は、オペレーションコードに従った後述するその他の処理ステップ3603を行い終了する。ステップ3601においてオペレーションコードが“Get Service”の場合は、ステップ3602において問合わせに使用された接続形式IDとサブアドレス：サービスID対応表とから、問合わせに使用された接続形式IDを持つサービスIDのリストを作成する。ステップ3604において、サービスID：タスクタイプ対応表を参照し、上記リストの中から指定されたタスクタイプを有するもののリストを作成する。ステップ3605において、ステップ3604で作成したリストの各レコードを検査し、有効フラグ804が有効、有効ユーザリスト805に値がある場合は問合わせに使用したユーザIDがその値に該当しないレコードを、無効ユーザリスト806に値がある場合は問い合わせに使用したユーザIDがその値に該当するレコードをリストから削除する。ステップ3609において、パラメータに含まれている条件情報とリスト中のサービスIDに対応する各マネージャの属性表と比較して、条件に合致する値以外をリストから削除する。ステップ3606において、ステップ3605において変更されたリストのレコードの数をチェックし、レコード数が0の場合は、要求されたサービスは存在しないものとしてステップ3611においてエラー返信パケットを作成し、ステップ3613において返信パケットを送信して終了する。ステップ3606においてレコード数が0以外の場合は、ステップ3607においてレコード数が1であるかどうかをチェックする。レコード数が1の場合は、ステップ3612において得られたサービスIDを含む返信パケットを作成し、ステップ3613において返信パケットを送信して終了する。ステップ3607においてレコード数が1以外の場合は、ステップ3608において、リスト中のサービスIDに対応する各マネージャに現在の負荷状態を問い合わせる。各マネージャは処理中のジョブの数を負荷状態として返し、この中から最低の負荷を持つマネージャを選択する。ステップ3610において、選択されたマネージャーに対応するサービスIDを含む返信パケットを作成し、ステップ3613において返信パケットを送信して終了する。

【００９５】（ジョブの構造）図52は、各マネージャが管理するジョブの実体を保持するファイル（ジョブファイル）の内部構造を示すものである。ジョブファイルのファイル名はジョブ表（図20）のファイル名1502で保持されている。またこのファイルの構造は、各コントローラが管理するジョブの実体を保持するファイルの内部構造も示している。このファイル名はジョブキュー表（図42）のファイル名2903で保持されている。ジョブの実体は、属性ID3701、属性値サイズ3702および属性値3703の組を複数連続して持つことによって表されている。ジョブがデータを含む場合は、3707、3708、3709で示されるように属性IDとしてデータを表す値、属性値としてファイル名のサイズ、属性値としてドキュメントデータを保持しているファイルのファイル名を保持している。属性の中には、データの送信方法、データのフォーマット（使用されているPDLなど）、イベントの種類とそのイベントが発生した時にイベント通知を送信する宛先などの情報、およびジョブの種類に依存した情報：プリントジョブの場合にはコピー部数、フィニッシング処理指定、使用するフォントの指定、使用するフォームオーバーレイの指定、ジョブ管理設定情報であるジョブ実行優先度、ジョブ指定実行時刻、ジョブ残存限度時刻などが含まれる。

【００９６】（各マネージャーでのジョブスクリプト処理）図53と図54は、各マネージャにおけるジョブスクリプトの処理フローを表すものである。ジョブスクリプトは、図46で示されるコマンドパケットの一続きによって

構成されており、"Job Start"オペレーションコードで始まり、"Job End"オペレーションコードで終わるものとして規定されている。ジョブスクリプトを構成する各パケットは、図9で示されるサブアドレス：サービスID対応表によって示されるサブアドレスに投入され、図47で示されるコマンドパケット処理フローによって各マネージャに配布される。図53で示される処理フローは、各マネージャに配布されたコマンドパケットを処理して図52で示されるジョブファイルおよびデータファイルを作成する場合の処理フローである。各マネージャに配布されたオペレーションは、ステップ3801においてオペレーションコードがサポートされているオペレーションコードであるかどうかをチェックする。チェックは、各マネージャが属性表内に保持している「サポートしているオペレーション」属性（属性ID101）の値と比較することにより行う。チェックの結果、サポート外のオペレーションであった場合は、ステップ3816においてエラー返信パケットを作成し、これを送信して終了する。ステップ3801のオペレーションコードがサポートされているものであったならば、ステップ3802において現在ジョブ投入中であるかどうかをチェックする。チェックは、RAM302の所定領域に記憶されているジョブ投入中フラグ（ステップ3806で設定されるフラグ）が真であるかどうかを検査することにより行う。このジョブ投入中フラグは、デフォルトで偽が設定されている。そして、検査の結果、ジョブ投入中フラグが偽であり、ジョブ投入中でなければ、ステップ3804においてオペレーションコードが"Job Start"であるかどうかをチェックする。チェックの結果、オペレーションコードが"Job Start"で無い場合は、ステップ3816においてエラー返信パケットを作成し、これを送信して終了する。

【００９７】ステップ3804のチェックの結果OKならば、ステップ3805においてジョブIDを割り当ててJob Startオペレーションのオプションであるパラメーター（ジョブ実行優先度、実行開始時刻、残存限度時刻）に基づいてジョブ表（図20）に項目を追加し、ジョブファイルをDISK315に新規に作成する。この指定実行時刻、残存限度時刻は、前述したように、相対時刻或いは絶対時刻で指定されるが、相対時刻で指定実行時刻、残存限度時刻が指定された場合には、このステップ3805においてその時の現在の時刻と指定された相対時刻から絶対時刻をCPU301が計算して、図20のジョブ表に登録される。その後ステップ3806において、ジョブ投入中フラグを真に設定してコマンドパケットの処理を終了する。ステップ3802のチェックにおいて、ジョブ投入中フラグが真の場合は、ステップ3803においてオペレーションコードが"Send"であるかどうかをチェックする。オペレーションコード"Send"は、パラメータにジョブを構成するデータが含まれていることをマネージャに指示するものである。ステップ3803においてオペレーションコードが"Send"の場合は、ステップ3807においてコマンドと共に入力された継続フラグをチェックし、継続フラグが真の場合はステップ3808において既に存在しているデータファイル3713に新たに領域を追加して、3809においてその領域にパラメータを書き込み終了する。ステップ3807において継続フラグが偽の場合は、ステップ3810においてデータファイル3713をDISK315に新規に作成し、ステップ3811においてパラメータをデータファイルに書き込む。その後、ステップ3812においてジョブファイルに領域を追加し、ステップ3813においてデータを表す属性ID3707、ファイル名のサイズ3708および新規作成したファイルのファイル名3709をこの領域に書き込み、終了する。ステップ3803においてオペレーションコードが"Send"でない場合、ステップ3814においてオペレーションコードが"Job End"であるかどうかをチェックする。オペレーションコードが"Job End"である場合は、ステップ3815においてジョブの生成終了処理を行い終了する。ジョブの生成終了処理には、ジョブファイルのクローズなどの処理を含み、マネージャの種類（プリントジョブ、スキャンジョブ、コピージョブ、フォント、フォームオーバーレイ、ログ、カラープロファイル）によって異なる。

【００９８】ステップ3814において、オペレーションコードが"Job End"でない場合は、ステップ3817においてオペレーションコードが"Set Job"であるかどうかをチェックする。オペレーションコード"Set Job"は、パラメータにジョブを構成する属性（属性IDと属性値）が含まれていることをマネージャに指示するものである。ジョブのオペレーションコードが"Set Job"である場合は、ステップ3818においてジョブファイルに領域を追加し、ステップ3819において属性ID、属性値サイズおよび属性値を追加した領域に書き込み終了する。

【００９９】ステップ3817において、オペレーションコードが"Set Job"でない場合は、ステップ3820においてオペレーションコードが"Send Request"であるかどうかをチェックする。オペレーションコードが"Send Request" は、データの送信をマネージャに指示するものであり、パラメータとしてデータの送信方法が含まれている。オペレーションコードが"Send Request"である場合は、ステップ3821において各マネージャに依存した処理を行い終了する。ステップ3821における処理は、スキャンジョブマネージャの場合は、ジョブファイルに保持されている属性値に従った原稿のスキャンをスキャナコントローラに指示し、得られたデータを指定された方法で送信すること、フォントマネージャの場合はジョブファイル内に属性値として指定されたフォントデータを即座に指定された送信方法で送信することなどが含まれる。ステップ3820においてオペレーションコードが"Send Request"でない場合は、ステップ3822においてその他のオペレーションコードの処理を行い終了する。

【０１００】（プリントジョブマネージャーでのジョブ

処理（1）：ジョブデータの受信からジョブスケジューリング表への登録）図55は、図５で示したプリントジョブマネージャ501から509、513におけるジョブの処理フローのうち、ジョブ受信からジョブスケジューリング表あるいは実行時刻順ジョブキュー表へのジョブの登録までの部分の処理を表したものである。プリントジョブマネージャが図54に示すジョブスクリプトの処理フローを行い、”Job Start”オペレーションコードが入力されることによりステップ3805でジョブ表に新たな項目が追加された後、CPU301は、図55で示す処理フローを実行する。ステップ3901においてデータ（ドキュメントデータ）の表現に使用されているPDL（Page Description Language）の種類を示す属性がジョブファイルの中に追加されるのを待ち、PDLの種類が確定すると使用するPDL Rasterizer（PDL Rasterizer417或いはPDL Rasterizer418）が使用可能となるまで待つ。ステップ3901でPDL Rasterizerが使用可能となると、ステップ3902においてデータ受信方法を示す属性がジョブファイルの中に追加されるのを待つ。データ受信方法を示す属性がジョブファイルの中に追加されると、ステップ3903においてデータの受信方法をチェックする。データの受信方法が多機能周辺機器MFP201の外部（例えば、ネットワーク上のClient PC等）にアクセスする方法である場合は、ステップ3904において指定されたデータの読込みを行うタスクを生成して外部にアクセスする。その後ステップ3905においてデータの読込みが開始されるのを待つ。ステップ3903において、データがジョブ内に含まれる場合（図54において“Send”オペレーションコードによるデータ受信の場合）、ステップ3906においてデータの受信が開始されるのを待つ。ステップ3905または3906のステップにおいてデータの読込みまたは受信が開始された時、ステップ3907においてデータをPDL Rasterizerに投入を開始する。その後、ステップ3908においてPDL Rasterizerからイメージデータを受け取る。ステップ3909においてこのジョブに指定実行時刻オプションが指定されている否かが図20に示すジョブ表から判断される。もし、指定されているならばジョブは図83に示すジョブスケジューリング表にではなく図84に示す実行時刻順ジョブキュー表にリンクされる（ステップ3911）。この際、同時に残存限度時刻の指定がされているかも図20に示すジョブ表から判断され、もし指定されているならば指定実行時刻と残存限度時刻の比較が行われ後者のほうが前者より大きい（時間的に後）などの矛盾した設定が成されている場合には表に登録されずにそのジョブは捨てられる。またステップ3909において指定実行時刻オプションが指定されていない場合にはステップ3910でジョブはジョブスケジューリング表の指定された実行優先度の最後尾にリンクされる。もしジョブ投入時に実行優先度が指定されていない場合にはデフォルトの優先度の３（最低優先度）が与えられる。

【０１０１】（プリントジョブマネージャのジョブ処理(2)：実行時刻順ジョブキュー表からジョブスケジューリング表への登録）プリントジョブマネジャは定期的に実行時刻順ジョブキュー表を監視するタスク（以後、実行時刻監視タスクと呼ぶ）を走らせており、該表中のジョブが指定された実行時刻に達したことを検出してジョブスケジューリング表に登録する処理を行わせている。図86に実行時刻監視タスクの処理を示す。ステップ3920において実効時刻順ジョブキュー表にリンクされている全ジョブについて検索が終了したかチェックする。チェック終了ならば処理は終了である。まだリンクしている全てのジョブについて検索が終了していなければステップ3921においてリンクの次のジョブをたどりその実行時刻を取得する。ステップ3921の処理は表の先頭ジョブから検索を開始しループを回るたびに検索対象のジョブを指すポインタを更新することで一度検索したジョブは重複して検索しない。ステップ3921において取得したジョブの指定実行時刻を現在時刻と比較する。ジョブの指定実行時刻が現在時刻を過ぎている場合はそのジョブは実行されるべき状態にあるのでステップ3923にてリンクから削除され引き続きステップ3924にてジョブスケジューリング表に優先度１で追加される。実行時刻順ジョブキュー表の全てのジョブについて実行時刻と現在時刻の比較が行われるまで本処理は続くことになる。全ジョブについての比較が終了したら実行時刻監視タスクは終了するが本タスクは時刻の経過を監視するのに十分短い周期で起動されるようになっている。

【０１０２】（プリントジョブマネージャのジョブ処理(3)：ジョブスケジューリング表から実行終了まで）プリントジョブマネジャは定期的にジョブスケジューリング表を監視するタスク（以後、ジョブスケジューラタスクと呼ぶ）を走らせており、該タスクは該表に新たなジョブが追加されたことあるいはジョブの優先度が変更されたことを検出してジョブスケジューリング表から次に実行されるべきジョブを取り出してプリントコントローラに投入する図87に示す処理を行っている。ジョブスケジューラタスクは図87のステップ3930からステップ3932の処理によって、ジョブスケジューリング表に登録（リンク）されているジョブのうち最も優先度の高くかつリンクの先頭のジョブを取り出す処理を行っている。具体的にはステップ3930においてプリントジョブマネージャのジョブスケジューリング表の検索対象優先度を１としてステップ3931において検索対象優先度の範囲がジョブスケジューリング表の優先度の範囲内（本実施例では優先度の範囲は１から３である）であるか否かをチェックする。範囲外であれば該表中には実行待ちのジョブは登録されていないことを意味するのでそのまま終了する。ステップ3932ではある優先度にリンクされているジョブのうち先頭にリンクされているジョブを探す処理である。もし存在しなければステップ3933において検索対象

優先度をインクリメントしてステップ3931に戻り再検索する。ステップ3932においてジョブが見つかると、このジョブがプリントコントローラに投入されるべきジョブとして確定する。ステップ3934にて検出されたジョブを表から削除し、引き続きステップ3935にて投入するプリントコントローラを選択する。この決定はプリントジョブマネージャー501から507と509と513の属性表の属性ID2001（Jobを実行するController IDリスト）で予め決められている場合と、プリントジョブマネージャー508の属性ID2002（Controller自動選択可能か否か）で可が指定されてプリントジョブファイルの内容（図52に示すジョブファイルの属性Iと属性値）を読むことによりプリントに必要となるプリンタコントローラを動的に決定する場合がある。この場合には、例えば、ジョブファイル内の属性IDと属性値でカラー印刷を指定されていればInk Jet Contorller512（CID23）を決定するし、属性IDと属性値でFinisherの使用が指定されていればLBP Controller510（CID21）を決定する。ステップ3936において選択した（単一あるいは複数の）プリントコントローラに対してステップ3934にて取り出したジョブを投入する。投入後はステップ3937においてプリントジョブマネジャのジョブ依頼表にジョブIDと各プリントコントローラID及び各プリントコントローラから受け取ったジョブIDをレコードとして追加する。その後、プリントジョブマネージャはステップ3938で各プリンタコントローラでジョブ終了を待ち、ジョブ終了がプリンタコントローラから通知されると、ステップ3939においてジョブ依頼表から対応するレコードを削除する。3940において各プリンタコントローラに依頼した全てのジョブが終了したかどうかをチェックし、まだプリンタコントローラ内でジョブが残っている場合は、ステップ3938に戻りプリンタコントローラでのジョブ終了を待つ。ステップ3940において、依頼した全てのプリンタコントローラでのジョブが終了した場合、ステップ3941においてジョブ表（図20）から、ジョブのレコードを削除する。この時、図61で示されるイベント送信処理を行う。ステップ3942において、ジョブファイルの中を検査し、ステップ3943においてジョブ終了に関するイベント送信を指示する属性値があるかどうかをチェックする。もしなければ処理は終了するが、ある場合はステップ3944において属性値を読み出すことにより送信方法と送信宛先を取得する。ステップ3945においてイベントの送信パケットを作成し、指定された送信方法と送信宛先に対してイベントを送信する。ステップ3946において、ジョブファイル・データファイルの削除などの終了処理を行い、ジョブの処理は終了する。

【０１０３】（プリントジョブマネージャのジョブ処理(4)：残存限度時刻指定のあるジョブの削除）プリントジョブマネジャの属性表の一つのジョブ表にはジョブ投入時にパラメータとして残存限度時刻が指定されている場合がある。これは投入されてからジョブの実行までに時間を要する場合に機密上大切なジョブを未実行状態で機器中に保持したくない場合に指定する属性である。すなわちこの残存限度時刻が指定されたジョブはプリントマネジャが定期的にチェックし限度時刻を超えて未実行状態にあるものは削除する。未実行ジョブはジョブスケジューリング表中にあるか実行時刻順ジョブキューもしくは各プリントコントローラのジョブキュー中に存在するが、本実施例では登録されているジョブスケジューリング表中にあるジョブで残存限度時刻が設定されているものについてのみ現在時刻との比較を行い限度を超えているジョブは該表中から削除してしまう処理を行う。プリントジョブマネジャはこのような機能の実現のために定期的にタスク（残存監視タスク）を起動している。図88はこの残存監視タスクの処理フローを示したものである。ステップ3950はジョブスケジューリング表の実行優先度の高いジョブから検索を始めて各ジョブのうち残存限度時刻が設定されているものについてその時刻と現在時刻の比較をおこなう（ステップ3951）。もし、この比較において、残存限度時刻を超えているものが発見されるとステップ3952においてそのジョブは表から削除されかつ続くステップ3953において対応するプリントジョブマネジャのジョブ表からも削除される。この処理をジョブスケジューリング表の全ジョブに対して行い残存監視タスクは自ら終了する。プリントジョブマネジャはまた一定時間経過後に残存監視タスクが起動されるようにシステムに設定を行っている。

【０１０４】（割込みプリントジョブマネージャのジョブ処理）プリントジョブマネジャのうち割り込みジョブを処理できるものを特に割込みプリントジョブマネジャと呼んでいる。図5の513が割込みプリントジョブマネジャである。このことは該プリントジョブマネジャ513がサービスID=19であり、図11からタスクタイプ=11であること、さらにタスクタイプ=11は図12から割込みプリントであることがわかる。割込みプリントジョブマネジャも他のプリントジョブマネジャと同様にプリントジョブマネージャが図54に示すジョブスクリプトの処理フローを行い、投入ジョブがジョブ表に追加されると図89の処理が行われる。まずステップ3960において既に割込みジョブの処理が行われている途中か否かがチェックされる。割込みプリントジョブマネジャは割込みジョブの多重投入を許さない仕様となっている。従って既に割込み処理中であればステップ3962においてジョブの受け取りを拒否するようコマンドパケットの戻り値を記入し発行元に返して終了する。割込み処理中でなければステップ3961において割込み処理中フラグをセットし多重投入をブロックする。ステップ3963からステップ3970は図55のステップ3901からステップ3908で示される通常のプリントジョブマネジャのジョブ処理と全く同一の処理を行いイメージデータを受け取る。ステップ3971ではジョブス

ケジューリング表や実行時刻順ジョブ表に登録することなくそのままジョブの処理を続ける。ステップ3971にて使用するプリントコントローラを決定しステップ3972にて選択したプリントコントローラにジョブを割込み指示と共に投入する。通常のプリントジョブマネジャはプリントコントローラに対してジョブを投入する際には割込み指示を与えない。この点が大きく割り込みプリントジョブマネジャと通常のプリントジョブマネジャの違いといえる。以後のステップ3973からステップ3982までは図87に示される通常のプリントジョブマネジャのジョブ処理のステップ3937からステップ3946と同一である。ジョブを終了した最後のステップ3983において割込みフラグをクリアして終了する。

【０１０５】（プリントコントローラのジョブ処理）図90にプリントコントローラのジョブ処理フローを示す。プリントコントローラはプリントジョブマネジャやコピーマネジャからジョブを投入されてその処理を開始する。プリントコントローラは処理に際して、ジョブキューの変化を監視するタスクを走らせておりこのタスクがジョブキューにリンクしているジョブをひとつ取り出してプリントエンジンに対しての印字依頼を行い、ジョブを処理する。プリントエンジンへのジョブの依頼はプリントエンジンの処理できる単位（通常ページ単位）で与えられ、プリントエンジンとプリントコントローラの間でハンドシェイクを行いながら実行していく。ジョブキューにリンクしている全ジョブを処理し終えたタスクは新たなジョブの投入を待って待機する。さて、プリントコントローラに対するジョブの投入時にはパラメーターとして実行モード指定が与えられる。実行モードには先着順実行モードと割込み実行モードがある。割込み実行モードを指定してジョブを投入してくるのは割込みプリントジョブマネジャ513と割込みコピーマネジャ609のみである。図90のステップ6001においてジョブ投入のパラメーターである実行モードのチェックが行われる。割込み実行が指定されていない場合はステップ6010にてプリントコントローラのジョブキューに追加される。割込み実行が指定されている場合にはステップ6002にて現在他の割込みジョブを処理中か調べる。これはプリントコントローラの割込みジョブ実行中フラグをチェックすることでわかる。ステップ6003では該割込みジョブ実行中フラグをセットして他の割込みジョブを多重に処理しないようにブロックする。ステップ6004では現在プリントエンジンで実行中のジョブに関するコンテキスト（図85参照）をDISK315内のコンテキスト退避用エリアに退避すると同時にプリントエンジンとハンドシェイクしているタスクをサスペンドさせる。その後ステップ6005にて割込みジョブのためにコンテキストを初期化する。ステップ6006においてプリントエンジンに対してジョブをプリントエンジンの処理する単位で分割して投入する。投入した一単位の終了時にはプリントコントローラはコンテキストを更新し、これを割込みジョブが終了するまで繰り返す（ステップ6007）。割り込みジョブの実行が終了したらステップ6008において、退避してあった割り込まれたジョブのコンテキストを退避エリアから復帰しサスペンドさせていたタスクをリジュームさせ、さらにステップ6009にて割込みジョブ実行中フラグをクリアする。この後は再開したタスクが実行途中であったジョブを再開することになる。この間プリントエンジン及びハンドシェイクしているタスクは割込みジョブの実行が割り込んできたことを意識することはない。ステップ6002において現在他の割込みジョブを実行中であればステップ6011にてエラー返信パケットを作成しパケット発行元に対して該パケットを送信して終了する。

【０１０６】（スキャンジョブマネージャでのジョブ処理）図56は、スキャンジョブマネージャ419におけるジョブの処理フローを表したものである。スキャンジョブマネージャはジョブの投入が終了してからジョブの処理を開始する。このため、スキャンジョブマネージャは図54におけるステップ3815のジョブの終了処理として図56のジョブ処理を開始する。ステップ4001において、スキャナコントローラ420にジョブを投入しジョブIDを受け取る。ステップ4002においてジョブ依頼表（図21）に、ジョブIDとスキャナコントローラIDおよびスキャナコントローラ420から受け取ったジョブIDを記録する。ステップ4003において、スキャナコントローラ420からのジョブ終了を待ち、ジョブ終了がスキャナコントローラ420から通知されると、スキャンしたイメージデータをスキャナコントローラ420から受け取り、ステップ4004においてジョブ表（図20）からジョブのレコードを削除する。この時、図61で示されるイベント送信処理を行う。ステップ4005において、ジョブ中の属性の中からデータの送信方法を指示しているものを検索する。ステップ4006においてデータ送信方法がデータをスクリプトとして送信する指示であった場合は、ステップ4008においてデータを他の属性情報と共にスクリプトとして送信する。機器からのデータ送信処理は詳細が図62に示されている。ステップ4006においてデータを参照として送信する指示であった場合は、ステップ4007においてデータを機器内部に保存し、これに対する参照情報を他の属性情報と共にスクリプトとして送信する。ステップ4010においてジョブファイルの中を検査し、ステップ4011においてジョブ終了に関するイベント送信を指示する属性値があるかどうかをチェックする。もしなければ処理は終了するが、ある場合はステップ4012において属性値を読み出すことにより送信方法と送信宛先を取得する。ステップ4013においてイベントの送信パケットを作成し、指定された送信方法と送信宛先に対してイベントを送信する。ステップ4014において、ジョブファイル・データファイルの削除などの終了処理を行い、ジョブの処理は終了する。

【０１０７】（コピージョブマネージャーでのジョブ処理（1）：スキャンジョブの依頼からジョブスケジューリング表への登録）図57は、コピージョブマネージャ601から609におけるジョブの処理フローを表したものである。コピージョブマネージャはジョブの投入が終了してからジョブの処理を開始する。このため、ジョブマネージャは図53におけるステップ3815のジョブの終了処理として図57のジョブ処理を開始する。ステップ4101において、スキャナコントローラ420にジョブを投入しジョブIDを受け取る。ステップ4102においてジョブ依頼表（図21）に、ジョブIDとスキャナコントローラIDおよびスキャナコントローラ420から受け取ったジョブIDを記録する。ステップ4103において、スキャナコントローラ420からのジョブ終了を待ち、ジョブ終了がスキャナコントローラ420から通知されると、スキャンしたイメージデータをスキャナコントローラ420から受け取り、ステップ4104においてジョブ依頼表から（図21）から、ジョブのレコードを削除する。ステップ4105においてこのジョブに指定実行時刻オプションが指定されている否かが図20に示すジョブ表から判断される。もし、指定されているならばジョブはジョブスケジューリング表にではなく実行時刻順ジョブキュー表にリンクされる（ステップ4106）この際同時に残存限度時刻の指定もされているかもチェックされ、指定されているならば指定実行時刻と残存限度時刻の比較が行われ後者のほうが前者より大きい（時間的に後）などの矛盾した設定が成されている場合には表に登録されずにそのジョブは捨てられる。またステップ4105において指定実行時刻オプションが指定されていない場合にはステップ4107でジョブはジョブスケジューリング表の指定された実行優先度の最後尾にリンクされる。もしジョブ投入時に実行優先度が指定されていない場合にはデフォルトの優先度の３（最低優先度）が与えられる。

【０１０８】（コピージョブマネージャのジョブ処理(2)：実行時刻順ジョブキュー表からジョブスケジューリング表への登録）コピージョブマネジャは定期的に実行時刻順ジョブキュー表を監視する実行時刻監視タスクを走らせており、該表中のジョブが指定された実行時刻に達したことを検出してジョブスケジューリング表に登録する処理を行わせている。実行時刻監視タスクの処理を示す。コピージョブマネージャの実行時刻監視タスクの処理は図86に示されるプリントジョブマネージャーの実行時刻監視タスクの処理フローと同一であるので説明は省略する。

【０１０９】（コピージョブマネージャのジョブ処理(3)：ジョブスケジューリング表から実行終了まで）コピージョブマネジャは定期的にジョブスケジューリング表を監視するジョブスケジューラタスクを走らせており、該タスクは該表に新たなジョブが追加されたことあるいはジョブの優先度が変更されたことを検出してジョブスケジューリング表から次に実行されるべきジョブを取り出してプリントコントローラに投入する処理を行っている。処理は図87に示すプリントジョブマネージャのジョブスケジューラタスクの処理フローと同一であるので説明は省略する。

【０１１０】（コピージョブマネージャのジョブ処理(4)：残存限度時刻指定のあるジョブの削除）コピージョブマネージャの属性表の一つのジョブ表にはジョブ投入時にパラメータとして残存限度時刻が指定されている場合がある。これは投入されてからジョブの実行までに時間を要する場合に機密上大切なジョブを未実行状態で機器中に保持したくない場合に指定する属性である。すなわちこの残存限度時刻が指定されたジョブはコピージョブマネージャが定期的にチェックし限度時刻を超えて未実行状態にあるものは削除する。未実行ジョブはジョブスケジューリング表中にあるか実行時刻順ジョブキューもしくは各プリントコントローラのジョブキュー中に存在するが、本実施例では登録されているジョブスケジューリング表中にあるジョブで残存限度時刻が設定されているものについてのみ現在時刻との比較を行い限度を超えているジョブは該表中から削除してしまう処理を行う。コピージョブマネージャはこのような機能の実現のために定期的にタスク（残存監視タスク）を起動している。コピージョブマネージャの残存監視タスクの処理は図88はに示されるプリントジョブマネジャの残存監視タスクの処理フローと同一であるので説明は省略する。

【０１１１】（割込みコピージョブマネージャのジョブ処理）コピージョブマネジャのうち割り込みジョブを処理できるものを特に割込みコピージョブマネジャと呼んでいる。図6の609が割込みコピージョブマネジャである。このことは該コピージョブマネジャ609がサービスID=20であり、図11からタスクタイプ=13であること、さらにタスクタイプ=13は図12から割込みコピーであることがわかる。割込みコピージョブマネジャも他のコピージョブマネジャと同様にコピージョブマネージャが図54に示すジョブスクリプトの処理フローを行い、投入ジョブがジョブ表に追加されると図91の処理が行われる。同図のステップ4130からステップ4132までは図89の割込みプリントジョブマネジャのジョブ処理のステップ3960からステップ3962と同一である。更に図91のステップ4133からステップ4136までは図57のコピージョブマネジャのジョブ処理のステップ4101からステップ4104までの処理と同一である。さらにステップ4137からステップ4149までは図89の割込みプリントジョブマネジャのジョブ処理のステップ3971からステップ3983と同一である。

【０１１２】その後、プリントジョブマネージャはステップ4108で各プリンタコントローラでジョブ終了を待ち、ジョブ終了がプリンタコントローラから通知されると、ステップ4109においてジョブ依頼表から終了したジョブに対応するレコードを削除する。ステップ4110にお

いて各プリンタコントローラに依頼した全てのジョブが終了したかどうかをチェックし、まだプリンタコントローラ内でジョブが残っている場合は、ステップ4108に戻りプリンタコントローラでのジョブ終了を待つ。ステップ4110において、依頼した全てのプリンタコントローラでのジョブが終了した場合、ステップ4111においてジョブ表（図20）から、ジョブのレコードを削除する。この時、図61で示されるイベント送信処理を行う。ステップ4112において、ジョブファイルの中を検査し、ステップ4113においてジョブ終了に関するイベント送信を指示する属性値があるかどうかをチェックする。もしなければ処理は終了するが、ある場合はステップ4114において属性値を読み出すことにより送信方法と送信宛先を取得する。ステップ4115においてイベントの送信パケットを作成し、指定された送信方法と送信宛先に対してイベントを送信する。ステップ4116において、ジョブファイル・データファイルの削除などの終了処理を行い、ジョブの処理は終了する。

【０１１３】（ジョブ処理ダウンロード）図58は、フォントマネージャ413、フォームオーバーレイマネージャ414、ログマネージャ415およびカラープロファイルマネージャ416におけるジョブの処理フローのうちデータのダウンロード機能に関するものを表すものである。これらのマネージャに対するジョブは、各マネージャが管理するデータのダウンロードおよびアップロードを行うためのものである。各マネージャが管理するデータの参照・削除などの管理は、Supervisor410が管理するサブアドレスに対してコマンドパケットを投入し図48で示されるように各マネージャの持つ属性表にアクセスすることによって行う。各マネージャはジョブの投入が終了してからデータダウンロードに関するジョブの処理を開始する。このため、各マネージャは図53におけるステップ3815のジョブの終了処理として図58のデータダウンロードに関するジョブ処理を開始する。ステップ4201において、ジョブファイル（図52）を走査し、データの受信方法に関する属性が存在するかどうかをチェックする。チェックの結果データの受信方法に関する属性が存在しない場合は、このジョブファイルはアップロードに関するもので図59に示す処理フローが既に行われているものとしてステップ4208において終了処理を行う。ステップ4208における終了処理ではジョブファイルの削除を行う。ステップ4201においてデータの受信方法に関する属性が存在する場合は、ステップ4202においてデータの受信方法をチェックする。データの受信方法がジョブ内に含まれる場合（図53において“Send”オペレーションコードによるデータ受信の場合）すでにデータが受信済みでありステップ4205において受信したデータをファイルとして保存する。データの受信方法が機器外部にあるデータである場合はステップ4203において指定された外部ソースにアクセスし、データの取得を行い、取得したデータをステップ4205においてファイルとして保存する。ステップ4205において保存したファイル情報を、ステップ4206において各マネージャが管理する管理表（フォントマネージャ413の場合フォント表（図28）、フォームオーバーレイマネージャ414の場合フォームオーバーレイ表（図31）、ログマネージャ415の場合ログ表（図34）、カラープロファイルマネージャ416の場合カラープロファイル表（図39）に新規レコードを書き込むことにより登録する。ステップ4207において、ジョブファイル・データファイルの削除などの終了処理を行い、ジョブの処理は終了する。

【０１１４】（ジョブ処理アップロード）図59は、フォントマネージャ413、フォームオーバーレイマネージャ414、ログマネージャ415およびカラープロファイルマネージャ416におけるジョブの処理フローのうちデータのアップロード機能に関するものを表すものである。各マネージャはジョブスクリプトを構成する“Send Request”オペレーションコードが投入された時点で、データアップロードに関するジョブの処理を開始する。このため、各マネージャは図54におけるステップ3821の各マネージャに依存した処理として図59のデータアップロードに関するジョブ処理を開始する。ステップ4301において、ジョブファイル中（図52）の属性の中からデータの送信方法を指示しているものをチェックする。ステップ4301においてデータ送信方法がデータをスクリプトとして送信する指示であった場合（NO）は、ステップ4303において、ジョブ内で指定データされたデータを他の属性情報と共にスクリプトとして送信する。機器からのデータ送信処理は詳細が図62に示されている。ステップ4301においてデータを参照として送信する指示であった場合は、ジョブ内で指定されたデータに対する参照情報を他の属性情報と共にスクリプトとして送信する。

【０１１５】（各マネージャーでのジョブ管理(1)）図60は、多機能周辺機器201内のプリントジョブマネージャ501から509、スキャンジョブマネージャ419およびコピージョブマネージャ601－608が管理するジョブに対する操作指示（ジョブの削除）をClientが行ったときに行われる処理フローを表したものである。各マネージャが管理するジョブに対する操作は、Supervisorの有するサブアドレスに対して適当なコマンドパケットを送信することにより行う。ジョブ管理用コマンドパケットにはパラメータとして、対象ジョブマネージャを特定するためのサービスIDおよび対象ジョブIDが含まれている。Clientから多機能周辺機器201に送られたコマンドパケットは、図47に示すフローによって処理され、Supervisor410に配布される。図60に示す処理フローは、図51で示したフロー中の3603（その他のOperation Codeの処理）のステップを詳細化したものである。ステップ4401，ステップ4403、ステップ4405、ステップ4407のおのおのにおいて、Supervisor410はオペレーションコードがジョブ

削除用のもの（”Cancel Job”）であるか、ジョブ優先度変更用のもの（“Promote Job”）であるか、ジョブ実行時刻変更用のもの（”Change Exec Time”）であるかあるいはジョブ残存限度時刻変更用のもの（“Change ExpireTime”）かどうかチェックし該当する処理（それぞれステップ4402のジョブ削除処理、ステップ4404のジョブ優先度変更処理、ステップ4406のジョブ実行時刻変更処理、ステップ4408の残存限度時刻変更処理）が行われる。オペレーションコードが上記のいずれでもない場合（ステップ4407でNoの場合）は、オペレーションコードに従った後述するその他の処理ステップ4409を行い終了する。全てのジョブマネジャが上記の処理を用意しているわけではない。たとえばスキャンジョブマネジャ419はジョブ優先度変更処理、ジョブ実行時刻変更処理、残存限度時刻変更処理は内容がない空の処理となっている。

【０１１６】（各マネージャーでのジョブ管理(2)：ジョブ削除処理）図60におけるステップ4402のジョブ削除処理は図92に詳細が示されている。図92のステップ4421においてパラメータ内に指定されたサービスIDに従って、パラメータをサービスIDに対応する各マネージャに送る。各マネージャはステップ4422において、指定されたジョブIDを各マネージャが管理するジョブ依頼表（図21）の中を検索する。ステップ4423においてジョブ依頼表の中に指定されたジョブIDがある場合は、ステップ4424において指定されたジョブIDに対応するコントローラのジョブIDとジョブを実行しているコントローラのコントローラIDを取得する。ステップ4425において、コントローラIDに対応するコントローラに対してコントローラのジョブIDを指定してジョブの削除を指示する。ステップ4426においてジョブ依頼表の中から該当するレコードを削除し、続いてステップ4433にてジョブ表（図20）から該当するジョブIDのジョブのレコードを削除しステップ4434にてジョブの削除が成功した旨の返信パケットを作成して送信する。ステップ4423にてジョブ依頼表中に指定されたジョブが発見できなかった場合には、ステップ4427でジョブスケジューリング表（図83）から探すことになる。ジョブスケジューリング表中に指定のジョブが見つかった場合には（ステップ4428）ステップ4429においてジョブスケジューリング表から指定ジョブを削除しステップ4433へ進む。ステップ4428にてジョブがジョブスケジューリング表中に指定されたジョブが発見できなかった場合には、ステップ4430で実行時刻順ジョブキュー表（図84）から探すことになる。実行時刻順ジョブキュー表中に指定のジョブが見つかった場合には（ステップ4431）ステップ4432において実行時刻順ジョブキュー表から指定ジョブを削除しステップ4433へ進む。ステップ4431にて指定されたジョブが発見できなかった場合は既にコントローラでのジョブが終了しているものとして、指定ジョブは存在しなかった旨の返信パケットを作成して送信する。

【０１１７】（各マネージャーでのジョブ管理(3)：ジョブ優先度変更処理）図60におけるステップ4404のジョブ削除処理は図93に詳細が示されている。図93のステップ4451においてパラメータ内に指定されたサービスIDに従って、パラメータをサービスIDに対応する各マネージャに送る。各マネージャはステップ4452において、指定されたジョブIDを各マネージャが管理するジョブスケジューリング表（図83）の中から検索する。ステップ4453においてジョブスケジューリング表中に指定されたジョブIDがある場合は、ステップ4454においてジョブスケジューリング表における指定ジョブのリンク位置を変更後の優先度にリンクしなおす。さらにステップ4457にてジョブ表（図20）中の指定されたジョブIDのジョブ優先度を変更しステップ4458にて変更終了の旨を記した返信パケットを作成して送信する。ステップ4453にてジョブスケジューリング表中に指定されたジョブIDが発見できなかった場合は、ステップ4455において実行時刻順ジョブキュー（図84）中から指定のジョブを検索する。ステップ4456において実行時刻順ジョブキュー表中に指定のジョブが見つかった場合にはステップ4457に進む。ステップ4456においても指定ジョブが発見できなかった場合には、指定ジョブは既にコントローラに投入された後か既に実行が終了してしまったかであるからステップ4458にてその旨を記した返信パケットを作成し、送信する。

【０１１８】（各マネージャーでのジョブ管理(4)：実行時刻変更処理）図60におけるステップ4406の実行時刻変更処理は図94に詳細が示されている。図94のステップ4471においてパラメータ内に指定されたサービスIDに従って、パラメータをサービスIDに対応する各マネージャに送る。実行時刻の変更が有効なのはジョブがまだ実行時刻順ジョブキュー（図84）中に存在する間であるからステップ4472にて実行時刻順ジョブキュー中に指定のジョブが存在するか検索する。ステップ4473にて指定ジョブが発見された場合にはステップ4474においてジョブ表中の指定ジョブの実行時刻を変更する。実行時刻は当初の指定時刻に比べて進めることも遅らせることも可能である。ステップ4475において現在時刻と変更された実行時刻の比較が行われる。実行時刻に達してしっまった場合はステップ4476に進むことになる。ステップ4476では指定ジョブを実行時刻順ジョブキューから削除し代わりにステップ4477にてジョブスケジューリング表（図83）に登録することになる。この際の優先順位は最高優先順位の優先度=1としてリンクされる。その後ステップ4478にて実行時刻変更処理が成功した旨の返信パケットを作成し送信する。ステップ4475においてまだ実行時刻に達していない場合にはジョブの移動は発生せず、そのままステップ4478に進む。ステップ4473にて指定ジョブが実行時刻順ジョブキュー中に存在しなかった場合にはステップ4478にて指定ジョブは発見できなかった旨を記した

返信パケットを作成し送信する。

【０１１９】（各マネージャーでのジョブ管理(5)：残存限度時刻変更処理）図60におけるステップ4408の実行時刻変更処理は図95に詳細が示されている。図95のステップ4481においてパラメータ内に指定されたサービスIDに従って、パラメータをサービスIDに対応する各マネージャに送る。残存限度時刻の変更が有効なのはジョブがまだ実行時刻順ジョブキュー（図84）中に存在するかジョブスケジューリング表（図83）中に存在する間である。したがってステップ4482、ステップ4483、ステップ4489、ステップ4490において両表のいずれかに指定されたジョブが存在するか検索する。両表のいずれかに指定ジョブが発見された場合にはステップ4484にてジョブ表中指定ジョブのレコードの残存限度時刻を変更する。両表のいずれにも発見されなかった場合にはステップ4488にて指定ジョブは発見されなかった旨を記した返信パケットを作成し送信する。ステップ4485において現在時刻と変更された後の残存限度時刻を比較する。既に残存限度時刻を過ぎていたらステップ4486に進み指定ジョブを削除することになる。ジョブは実行時刻順ジョブキュー表かジョブスケジューリング表中のいずれかに存在するの。さらにステップ4487にてジョブ表から該当するジョブIDのジョブのレコードを削除しステップ4488にてジョブの削除が成功した旨の返信パケットを作成して送信する。

【０１２０】（イベント送信）図61は、各マネージャにおけるイベント送信の処理フローを表したものである。各マネージャは図16に示す様なイベント設定表を、各自が保持する属性表の値として持っている。図16の説明で示したように、この表にはイベントが発生したときにイベントを送信する接続形式と宛先とが書かれている。あるイベントが発生すると、各マネージャはイベントに対するイベントIDを認識する。その後、ステップ4501において、イベント設定表を参照しイベントIDが登録されているかどうかを検索する。ステップ4502において、イベントIDが１つも登録されていなければ処理を終了する。ステップ4502においてイベントIDが１つでも登録されていれば、ステップ4503でイベントを送信する際に使用する接続形式とイベントを送信する宛先を最初のレコードから取得し、ステップ4504においてイベント送信パケットを作成する。このパケットの中にはイベントID毎に規定されているパラメータと、Supervisor410が保持するイベントフォーマット表（図17）に登録されているイベントID毎のフォーマットに従ったパラメータとを付加する。ステップ4505において、このイベント送信パケットを4503で取得した接続形式の送信宛先に送信する。そして、ステップ4506でイベントIDが有るレコード全てについてステップ4503からステップ4505までを繰り返し処理したか否かを判断し、処理していなければステップ4503に戻り、全てのレコードが処理されていれば終了する。

【０１２１】（多機能周辺機器からのデータスクリプト送信）図62は、多機能周辺機器201からデータをコマンドパケットの連続であるスクリプトとして送信する際の処理フローを表す。この処理フローは、スキャンジョブの結果得られるイメージデータの送信ステップ4008やフォントデータの送信ステップ4303などにおいて使用されるものである。ステップ4601において、データの属性を含むパケットを生成して送信する。必要ならステップ4501を繰り返し、複数の属性を送信する。ステップ4602において、送信を指定されたデータを取得する。コマンドパケットは図46に示す構造を有し、パラメータとして送信できるサイズがかぎられているため、4603において、データの長さを検査する。検査の結果、データの長さが制限値の長さ（64Kバイト）を超えている場合、ステップ4606においてデータを制限値の長さで切断し、得られたデータをステップ4607においてパラメータとして付加したコマンドパケットを生成し送信する。このコマンドパケットには、オペレーションコードとして“Send”を設定し、継続フラグを真として設定する。ステップ4608において、切断した残りのデータを取得しステップ4603のデータ長の検査を再び行う。ステップ4603の検査の結果、データ長がコマンドパケットの制限値の長さに入っている場合は、ステップ4604においてデータをパラメータとして付加したコマンドパケットを生成し送信する。このコマンドパケットには、オペレーションコードとして“Send”を設定し、継続フラグを偽として設定する。ステップ4605において、残りの属性を含むコマンドパケットを生成して送信する。必要ならステップ4501を繰り返し、複数の属性を送信して終了する。

【０１２２】（Client PCのハード構成）図63は図2に示す本実施形態のClientPC202,203,204,205の共通したハードウエア構成（コントローラ）を示す図である。コントローラの内部では、CPU6001がバス6010を介して、Memory(RAM)6002，CRT等のディスプレイ6003、キーボードやマウス等のポインティングデバイス6004，ROM6008，DISK6009が接続されている。図77に示す各種プログラム及びデータは、ハードディスクやフロッピーディスク等のDISK6009（記憶媒体）に記憶されており、必要に応じて順次Memory(RAM)6002に読み出されてCPU6001で実行される。このDISK6002は、Client PCに着脱可能でもClient PCに内蔵されたものでも良い。更に、図77に示すプログラムは、ネットワークインタフェースケーブル208（10BASE-T）、IEEE1394インタフェースケーブル206、 IEEE1284インタフェースケーブル207を介して他のClient PC、MFP201からダウンロードされてDISK6009に記憶される構成でも良い。図63に示すハードウエアは図78に示す一般的なClient PCを構成する。CPU6001がディスプレイ6003にデータを書き込むことにより表示を行い、CPU6001がポインティングデバイス6004からデータを読み出すことにより、ユーザからの指示を入力する。

【０１２３】また、バス6010には、ネットワークインターフェースコネクタ6005、IEEE1394インターフェースコネクタ6006、IEEE1284インターフェースコネクタ6007が接続されており、図2に示すEthernet(10BASE-T)ケーブル208、IEEE1394ケーブル206、IEEE1284ケーブル207が接続される。CPU6001がこれらのインターフェースからデータを読み込みまたは書き込むことによりそれぞれのインターフェースを使用した通信を行う。

【０１２４】（Client PCのソフトウエア構成）図64は多機能周辺機器201を利用するClient202、203、204、205のソフトウエア（制御プログラム）ブロック図の一部を表す。Client PCが使用するソフトウエア（制御プログラム）とデータは図77に示すようにDISK6009に記憶されている。501は、ユーザインターフェースであり、これによって505から514で示される各ドライバおよびユーティリティーが多機能周辺機器201の持つ情報をディスプレイ6003に表示する。502は現在使用中の多機能周辺機器201の情報（データ）を保持するデーターベース（DISK6009）であり、Clientが多機能周辺機器201に接続した段階で、503の機器情報取得部が機器の持つ全ての情報を取得して502のデータベースの中に保持する。504は、多機能周辺機器201が保持する各属性表の中で使用される、各属性の意味、各属性ID毎のデータ型、イベントID毎に規定されているパラメータのフォーマット、タスクタイプの意味、Supervisorのサブアドレスなどの規定情報を保持している規定データベース（DISK6009）である。505から514で示される各ドライバおよびユーティリティーは、規定データベース504および機器情報データベース502に基づき動作する。5050は、プリンタドライバでありClient PC上で動作するアプリケーションプログラムからの指示によりプリントジョブスクリプトを生成する。506は、スキャナドライバでありＣｌｉｅｎｔ　ＰＣ上で動作するアプリケーションプログラムからの指示によりスキャンジョブスクリプトを生成する。５０７は、コピードライバでありClient PC上で動作するアプリケーションプログラムからの指示によりコピージョブスクリプトを生成する。508は、フォント管理ユーティリティであり、フォントデータのダウンロードおよびアップロードを行うジョブを生成し、フォントデータを管理するコマンドを生成する。509は、フォームオーバーレイ管理ユーティリティであり、フォームオーバーレイデータのダウンロードおよびアップロードを行うジョブを生成し、フォームオーバーレイを管理するコマンドを生成する。510は、ログ管理ユーティリティであり、ログデータのダウンロードおよびアップロードを行うジョブを生成し、ログを管理するコマンドを生成する。511は、カラープロファイル管理ユーティリティであり、カラープロファイルデータのダウンロードおよびアップロードを行うジョブを生成し、カラープロファイルを管理するコマンドを生成する。512は、ジョブ管理ユーティリティであり、プリントジョブ・スキャンジョブ・コピージョブの削除・一時停止・実行再開などの管理をおこなうコマンドを生成する。513は、機器管理ユーティリティであり、プリンタコントローラ・スキャナコントローラの状態を取得するコマンドを生成する。514は、課金管理ユーティリティであり、課金データの取得するコマンドを生成する。

【０１２５】515は、505から514で示される各ドライバおよびユーティリティーから生成されたコマンドを基にコマンドパケットを生成するジェネレータである。生成されたコマンドパケットは、517、519、521の各トランスポート処理モジュールを利用して多機能周辺機器201に送信する。ジェネレータ515は、517、519、521の各トランスポート処理モジュールから入力された返信パケットおよびイベントパケットを解釈し、適当なドライバまたはユーティリティに配布する。517は、IEEE1394のトランスポート層であるSBP－2の処理モジュールである。519は、TCP/IPおよびUDP/IPの処理モジュールである。521は、IEEE1284のトランスポート層であるIEEE1284.4の処理モジュールである。518は、IEEE1394の物理層を処理するインターフェースである。520は、ネットワークインターフェースである。522は、IEEE1284の物理層を処理するインターフェースである。図64はまた、多機能周辺機器201のソフトウエア（制御プログラム）ブロック図（図4）におけるユーザーインターフェースマネージャ405の内部構成を表すものでもある。ユーザインターフェースマネージャ405の場合は、図64におけるSBP－2処理モジュール517、TCP/IPおよびUDP/IP処理モジュール519、IEEE1284.4処理モジュール521、ネットワークインターフェース518、ネットワークインターフェース520、IEEE1284インターフェース522は存在せず、ジェネレータ515が図4におけるインタープリター409に直接接続されており、これによってコマンドパケット・返信パケット・イベントパケットのやり取りが行われている。この直接接続は他のClient202、203、204、205のソフトウエア（制御プログラム）構成には存在していない。

【０１２６】（パケット生成・送信処理）図65は、ジェネレータ515が505から514で示される各ドライバおよびユーティリティーからコマンドまたはジョブスクリプトを受けて、図46で示されるコマンドパケットを生成し多機能周辺機器201に送信する処理フローを表す。この処理フローの前に、送信先の多機能周辺機器の適当なサブアドレスには前もって接続されているものとする。どのサブアドレスに接続するべきかという情報は、図66で示される機器情報の取得フローによって取得される。ジョブスクリプトはコマンドの連続で定義されており、処理対象がジョブスクリプトの場合は、全てのコマンドが処理されるまでステップ4701からステップ4706までを繰り返す。以下の記述において、ジョブスクリプトの発行とはステップ4701からステップ4707まで、各コマンド毎に

処理することを意味する。ステップ4701において、コマンドと共に送信するデータの長さを検査する。検査の結果、パラメータの制限値の長さ（64Kバイト）を超える場合は、ステップ4702においてデータを制限値の長さで切断する。切断されたデータはステップ4703において、継続フラグを真としてコマンドに対応したオペレーションコードを含むパケットを生成し、ステップ4704においてパケットを送信する。ステップ4701において、データの長さがパラメータの制限値の長さに入る場合は、ステップ4705において継続フラグを偽としてコマンドに対応したオペレーションコードを含むパケットを生成し、ステップ4706においてパケットを送信する。

【０１２７】（Clientでの情報取得とソフトウエア自動構成）図66は、図64の機器情報取得部503の処理フローチャートを表す。機器情報取得部503はClientが多機能周辺機器201と接続直後に図66で表す処理を行い、機器情報を取得して機器情報データベース502（DISK6009）に保持する。この処理はまた、機器からコンフィギュレーション変更のイベントが送られたときにも再度動作する。ステップ4801において、Client PCは多機能周辺機器201と接続する。具体的には、Ethernet208に接続されているClientPC202,203はIPアドレス、ポート番号を指定して多機能周辺機器201に接続しIPパケットデータを出力する。IEEE1394インタフェース206に接続されているClientPC204はノードIDとLUN（ロジカルユニットナンバー）を指定して多機能周辺機器201と接続しSBP−2パケットデータを出力する。IEEE1284インタフェース207に接続されているClientPC205はソケット番号を指定して多機能周辺機器201と接続しIEEE1284．4パケットデータを出力する。接続先のサブアドレスは、接続に使用した接続形式ごとにあらかじめ規定されているSupervisor用のサブアドレスを使用する。この規定値は、規定情報データベース504（DISK6009）から取得する。ステップ4802において、図7に示すSupervisorの属性表の属性ID100（属性IDのリスト）の値を取得する。属性値の取得は、SupervisorのサービスID0と属性IDをパラメータとして属性値取得用コマンド（Get）を生成し、ジェネレータ515にコマンドを投入することにより行う。ジェネレータ515は図65の処理フローを行い、属性値取得用のコマンドパケットを、多機能周辺機器201に送信する。多機能周辺機器201では、上記コマンドパケットを受信し、図47および図48の処理フローを行って指定された属性IDの属性値をClientに返信する。Clientでは返信パケットをジェネレータによって解析し、機器情報取得部に送る。尚、属性ID100の属性は、Supervisorの属性表にある全属性IDのリストとなっている。ステップ4803において属性IDを指定して属性値を取得し、ステップ4804で全ての属性IDの属性値を取得するまで繰り返し処理をし、その属性値を属性IDと共に機器情報データベース502に保存する。以上のステップにおいてSupervisorの持つ属性表（図7）が全て機器情報データベース502に保存される。

【０１２８】そして、ステップ4805において機器情報データベース502（DISK6009）に保存された属性表からサービスIDのリスト（図11に示すSID：TASK TYPE表）を取得する。なお、サービスIDの取得は、サービスIDのリスト問合わせ用コマンド（LISTSERVICE）を生成し、ジェネレータ515にコマンドを投入することにより行うこともできる。この時、多機能周辺機器201では、図47、図48、図49で示される処理フローが実行される。ステップ4806からステップ4811において、ステップ4805において取得した各サービスID毎に各マネージャの情報を取得する。ステップ4806で図11に示すリストからSupervisor（サービスID0）を除くサービスIDの中で最初のサービスIDであるサービスID1（プリントジョブマネージャ501）を取得する。ステップ4807でサービスIDに対応するサブアドレスの取得を行う。サブアドレスの取得は、サービスIDをパラメータとしてサブアドレス取得用のコマンド（Reserve）を生成し、ジェネレータ515にコマンドを投入することにより行う。この時、多機能周辺機器201では、図47、図48、図49、図50で示される処理フローが実行される。取得したサブアドレスは、機器情報データベース502に保存する。ステップ4808において、各マネージャの属性表の属性ID100の値（属性IDのリスト）を取得する。属性値の取得は、各マネージャに対応するサービスIDと属性IDをパラメータを指定して属性値取得用コマンド（Get）を生成し、ジェネレータ515にコマンドを投入することにより行う。ステップ4809において属性IDを指定して属性値を取得し、ステップ4810で全ての属性IDの属性値を取得するまで繰り返し処理をし、その属性値を属性IDと共に機器情報データベース502に保存する。ステップ4811においてサービスIDのリストの最後のサービスIDか否かを判断し、最後でなければステップ4812においてリストの次のサービスIDをリストから取得してステップ4807からの処理を繰り返す。ステップ4811においてサービスIDのリストの全てのサービスID（プリントジョブマネージャー501、502,503,504,505,506,507,508,509,スキャンジョブマネージャー419、コピーマネージャー601,602,603,604605,606,607,608,フォントマネージャー413、フォームオーバーレイマネージャー414、ログマネージャー415、カラープロファイルマネージャー416、プリンタコントローラ510,511,512,スキャナコントローラー420）の属性値を属性IDと共に機器情報データベース502に保存する。そして、ステップ4813で機器情報データベース502（DISK6009）に保存された機器情報（データ）に基づいて各ドライバー・ユーティリティー（制御プログラム）505から514のそれぞれに対応したマネージャーのためのユーザーインタフェースを自動構成する。自動構成ステップ4813の処理には、ユーザーインターフェースの更新が含まれており、ドライバ

ー・ユーティリティに応じて設定画面（ユーザーインタフェース）の表示を変える。例えば、プリンタドライバーのユーザーインタフェースは、ディスプレイ6003に表示される表示画面を示す図81や図82のように、各プリントジョブマネージャー（501から509）の属性ID1101の属性値に基づいてサポートしているPDLのリスト、属性ID1102に基づいてカラー印刷可能か、属性ID1103の属性値に基づいてサポートしているFinishingの種類、属性ID1104の属性値に基づいて設定可能最高解像度、属性ID1105に基づいて設定可能最低解像度の表示を変える。同様に他のドライバ506、507、ユーティリティ508から514のためのユーザーインタフェースの表示も属性に応じて変える。更に、各ドライバ・ユーティリティの（プリントマネージャ、コピージョブマネージャー等）のアイコンを識別可能に表示する。最後に、ステップ4814においてSupervisor410との接続を終了して、処理は終了する。

【０１２９】（タスクタイプを指定したジョブの発行）図67は、各アプリケーション・ユーティリティがタスクタイプを指定したジョブスクリプトを発行した場合の処理を表すものである。サービスIDを指定したジョブの発行は、サービスIDをキーとして機器情報データベース502を検索し、得られたサブアドレスに対して適切なジョブスクリプトを送信することにより行うが、ユーザからの指示によりタスクタイプのみが指定された場合は、多機能周辺機器201に対して使用するサービスIDの紹介依頼を指示する必要がある。図67は、この場合の処理を示すものである。ステップ4901おいて、タスクタイプを指定してサービスIDを取得する属性値取得用コマンドを生成し、ジェネレータ515にコマンドを投入することにより行う。多機能周辺機器201では、図47、図48、図49、図50、図51で示される処理フローが実行される。ステップ4902において、ステップ4901で得られたサービスIDを基に機器情報データベース502を検索し、得られたサブアドレスに対してジョブスクリプトを送信する。

【０１３０】（イベントの構造）図68は、多機能周辺機器201からClientに送られるイベントの構造を表す。イベントパケットは図46に示されるコマンドパケットと同一の構造を持ち、図68で示すものは、パケットのパラメータ3110の部分の構造である。5001はイベントの種類を表すイベントIDである。5002はイベントIDごとに規定されているパラメータデータである。このデータのフォーマットは規定情報データベース504に予め保持されている。5003はイベントIDごとに機器でそのフォーマットが決められているパラメータデータである。このデータのフォーマットはイベントフォーマット表（図17）として、図66の処理フローを実行することにより機器情報データベース502に保持されている。

【０１３１】（イベントの処理）図69は、Clientが多機能周辺機器201から各ドライバー・ユーティリティーがイベントを受けたときの処理を表すフローである。各ドライバー・ユーティリティーは、自分が欲しいイベントをイベントIDをキーとしてジェネレータ515に予め登録しておく。イベントが多機能周辺機器201からClientに送られると、ジェネレータ515は登録してある各ドライバー・ユーティリティーにイベントを配布する。図69は、各ドライバー・ユーティリティーがイベントの配布を受けた後の処理を表すものである。ステップ5101において、規定情報データベース504から得たフォーマット情報を元に規定パラメータデータ5002の解析を行う。ステップ5102において、イベントID（5001）をキーとして機器情報データベース502に保持されているイベントフォーマット表（図17）からイベントIDに対応するイベントフォーマット1302を取得する。ステップ5103において、取得したイベントフォーマットに基づき、機器依存のイベントフォーマット（イベントフォーマットの属性ID676：紙サイズ、ID756：紙の種類、ID666：トナーの種類、ID698：インクの種類、ID600：カバーの位置）を解析する。そして、ステップ5104において解析されたパラメータとイベントID （イベント ID200：紙なし、ID399：トナーなし、ID432：インクなし、ID234：多機能周辺機器のカバーオープン）を処理してイベントに応じた表示（例えば、紙なし、カバーオープン、トナーなし等のユーザーインタフェースをディスプレイ6003に表示する）を行う。

【０１３２】（ログの処理）図70は、ログ管理ユーティリティ510が多機能周辺機器201が保持するログを取得して処理するフローを表すものである。ステップ5201において、ログデータの取得を行う。ログデータの取得は、ログデータ取得用ジョブスクリプトを生成し、多機能周辺機器201に送ることにより行う。多機能周辺機器201では、図47、図53、図59で示される処理フローが実行され、ログデータがClientに送られる。例えば、図35に示すLogFormatID1、98/1/3 1:23のログデータ「12345,4,1.23,“OK”」は、サービスID1のプリントジョブマネージャーにジョブを発行したユーザーIDが12345で、出力した紙数が4で、使用したトナー量が1.23、ジョブの終了状態がOKを示す。ステップ5202において、機器情報データベース502からログフォーマット表（図36）を取得する。ステップ5203において、ログデータの各レコードの先頭にあるログフォーマットID2401を取得し、この値からログフォーマット表（図36）の中からログフォーマット2502を取得する。例えば、LogFormatID1のフォーマット「1：701,1：565,1：765,1：777」は、サービスID1のプリントジョブマネージャーにジョブを発行したユーザーID、出力した紙数、使用したトナー量、ジョブの終了状態を示す。また、LogFormatID2のフォーマット「11：701,11：565,11：765,11：777」は、サービスID11のプリントジョブマネージャーにジョブを発行したユーザーID、出力した紙数、使用したトナー量、ジョブの終了状態を示す。ステップ5204において、ログデータ24

03の解析をログフォーマットフォーマット2502に従って行い、解析したデータをログ発生時刻2402と共に処理（ログをソートする等）する。そして、ステップ5205において各レコードのログデータを解析したか否かを判断し、解析していなければステップ5203からの処理を繰り返し実行し、全て解析していれば処理を終了する。

【０１３３】（課金情報の取得・処理）図71は、課金管理ユーティリティ514が多機能周辺機器201が保持する課金データを取得して処理するフローをあらわす。課金データは、各マネージャが保持するカウントデータとして表される。カウントデータは属性ID401の値として属性表（図7、図18、図22、図24、図26、図29、図32、図37、図40、図44）に保持されており、その値は整数値のリストとなっている。各整数値の意味するところは、各マネージャの属性表の中にカウントデータフォーマットの属性（属性ID402）の値として表現されており、値は属性IDのリストとなっている。カウントデータのリストとカウントデータフォーマットのリストは、リスト中の順番によって対応している。例えば、カウントデータの3番目の値は、カウントデータフォーマットの3番目の属性IDの情報を意味している。図71は、各マネージャが保持するカウントデータを取得して処理するフローを表すものである。ステップ5301において、対象となるマネージャのサービスIDを指定して、属性ID401の値（カウントデータの値）を取得する。例えば、属性ID401の値（45,78,34,13）は、属性ID402のカウントデータフォーマットで表わされる用紙サイズのプリント枚数を示す。この属性ID401の値の取得は、サービスIDと属性IDをパラメータをパラメータとして属性値取得用コマンドを生成し、ジェネレータ515にコマンドを投入することにより行う。ステップ5302において、同じサービスIDを指定して、属性ID402の値（カウントデータフォーマットの値）を取得する。例えば、属性ID402の属性値565が用紙サイズA2のプリント枚数、属性値537が用紙サイズA3のプリント枚数、属性値545が用紙サイズA4のプリント枚数、属性値523が用紙サイズA5のプリント枚数を意味する。これは機器情報データベース502から取得してもよい。ステップ5303において、取得したカウントデータフォーマットを使用して、カウントデータの解析を行う。ステップ5304において、解析したカウントデータを使用して、例えば、サービスID毎、ユーザーID毎、部門毎に課金データの処理・生成を行って終了する。

【０１３４】（プリントジョブ、データダウンロードジョブの発行）図72は、プリンタドライバ505によるプリントジョブの発行、フォント管理ユーティリティ508によるフォントデータダウンロードジョブスクリプト発行、フォームオーバーレイユーティリティ509によるフォームオーバーレイデータダウンロードジョブスクリプト発行、ログ管理ユーティリティー510によるログデータダウンロードジョブスクリプト発行およびカラープロファイル管理ユーティリティ511によるカラープロファイルデータダウンロードジョブスクリプト発行の各処理を表す。以上の各ソフトウエア（制御プログラム）による処理フローは共通であり、プリント対象ドキュメントデータおよび上記各種類のダウロード対象データ（フォントデータ、フォームデータ、カラープロファイルデータ、ログデータ）は、図72および以下の説明において、「データ」と記述されている。ジョブ発行の際には、プリントまたはダウンロードするデータがジョブスクリプトの中に設定する複数の属性値と共に指定される。ステップ5401において、機器情報データベース502にアクセスして機器情報を取得する。ステップ5402において、データが投入対象とする各マネージャ（例えば、プリント対象ドキュメントデータなら印刷を指定するプリントマネージャ）の有する属性表を参照し、指定された属性値が属性表で設定されている各属性値の範囲内にあるかどうかをチェックする。チェックの結果、１つでも範囲内に無い場合は、ステップ5405においてユーザインターフェース上にエラーダイアログを表示するなど、ジョブ発行拒否の処理を行い、終了する。

【０１３５】ステップ5402のチェックの結果、全ての属性がマネージャの持つ属性表内の各属性値の範囲内にある場合は、ステップ5403においてマネージャの属性表中に禁止設定属性（属性ID801から805）が存在するかどうかチェックする。マネージャの属性表が禁止設定属性を持つ場合は、ステップ5404においてマネージャが属性表内に持つ各禁止設定属性毎に指定されたジョブの属性が禁止されている設定値の組み合わせとなっていないかどうかチェックする。チェックの結果、禁止されている属性の設定値の組み合わせがある場合は、ステップ5405においてジョブ発行拒否の処理を行い終了する。ステップ5403のチェックの結果、マネージャが禁止設定属性を持たない場合およびステップ5404のチェックの結果ジョブの属性が禁止されている設定値の組み合わせとなっていない場合、ステップ5406においてデータの存在場所をチェックする。データの存在場所の指定はは、ユーザーがアドレスを指定するかアプリケーションがアドレスを生成する事で行われる。データの存在場所がジョブを発行するClient内部の場合は、ステップ5407において機器情報データベース502内の各担当マネージャの属性表の属性値601（サポートしているデータのダウンロード方法）をチェックする。

【０１３６】ステップ5407のチェックにおいて、ジョブ内にデータを含ませる方法がサポートされている場合、ステップ5408においてジョブ内にデータを含ませたジョブスクリプトを生成し、図65に示すパケット生成処理を経て、多機能周辺機器201に送信する。ステップ5407のチェックにおいて、ジョブ内にデータを含ませる方法がサポートされていない場合、ステップ5409において、機器内に存在するデータの参照ポインタ（URL：Unified R esource Locator）をジョブ内に含ませたジョブスクリプトを生成し送信する。ステップ5406のチェックにおいて、データがジョブを発行するClient外部に存在する場合は、ステップ5410において機器情報データ内の各担当マネージャの属性表の属性値601（サポートしているデータのダウンロード方法）をチェックする。ステップ5410のチェックにおいて、ジョブ内に参照ポインタを含ませて機器自身がデータを読み込む方法がサポートされている場合は、データに対する参照ポインタをジョブ内に含ませたジョブスクリプトを生成し送信する。ステップ5410のチェックにおいて、ジョブ内に参照ポインタを含ませて機器自身がデータを読み込む方法がサポートされていない場合は、ステップ5412においてデータを機器外部から機器内部へ一旦読み込む。そして、ステップ5413において、読み込んだデータをジョブ内に含むジョブスクリプトを生成して送信する。ジョブスクリプトを受信した多機能周辺機器201の各マネージャは、図53、図54、図55（プリントジョブマネージャ）または図53、図54、図58（フォントマネージャ、フォームオーバーレイマネージャ、ログマネージャ、カラープロファイルマネージャ）に示される処理を行い、指示されたジョブの処理を行う。

【０１３７】（スキャンジョブ、データアップロードジョブの発行）図73は、スキャナドライバ506によるスキャンジョブの発行、フォント管理ユーティリティ508によるフォントデータアップロードジョブスクリプト発行、フォームオーバーレイユーティリティ509によるフォームオーバーレイデータアップロードジョブスクリプト発行、ログ管理ユーティリティー510によるログデータアップロードジョブスクリプト発行およびカラープロファイル管理ユーティリティ511によるカラープロファイルデータアップロードジョブスクリプト発行の各処理を表す。以上の各ソフトウエア（制御プログラム）による処理フローは共通であり、スキャン結果得られたドキュメントデータおよび上記各種類のアップロード対象データ（フォントデータ、フォームデータ、カラープロファイルデータ、ロゴデータ）は、図73および以下の説明において、「データ」と記述されている。ジョブ発行の際には、データを指し示す指示子が、ジョブスクリプトの中に設定する複数の属性値と共に指定される。ステップ5501において、機器情報データベース502にアクセスして機器情報を取得する。ステップ5502において、データをアップロード対象とする各マネージャ（例えば、フォントデータならフォントデータを管理するフォントマネージャ）の有する属性表を参照し、指定された属性値が属性表で設定されている各属性値の範囲内にあるかどうかをチェックする。チェックの結果、１つでも範囲内に無い場合は、ステップ5505においてユーザインターフェース上にエラーダイアログを表示するなど、ジョブ発行拒否の処理を行い、終了する。ステップ5502のチェックの結果、全ての属性がマネージャの持つ属性表内の各属性値の範囲内にある場合は、ステップ5503においてマネージャの属性中に禁止設定属性（属性ID801から805）が存在するかどうかチェックする。マネージャの属性表が禁止設定属性を持つ場合は、ステップ5504においてマネージャが属性表内に持つ各禁止設定属性毎に指定されたジョブの属性が禁止されている設定値の組み合わせとなっていないかどうかチェックする。チェックの結果、禁止されている属性の設定値の組み合わせがある場合は、ステップ5505においてジョブ発行拒否の処理を行い終了する。

【０１３８】ステップ5503のチェックの結果、マネージャが禁止設定属性を持たない場合およびステップ5504のチェックの結果ジョブの属性が禁止された設定値の組み合わせとなっていない場合、ステップ5506において、データの送信宛先（保存位置）をチェックする。データの送信宛先が機器外部の場合は、ステップ5507において、再度データの送信宛先をチェックする。データの送信宛先がジョブを発行するClientの場合は、ステップ5508において機器情報データ内の各担当マネージャの属性表の属性値602（サポートしているデータのアップロード方法）をチェックする。ステップ5508のチェックの結果データ送信をサポートしている場合は、ステップ5509において、データ送信を指示するジョブスクリプトを発行し、ステップ5510においてデータ受信を行う。ステップ5508のチェックの結果データ送信をサポートしていない場合（NO）は、ステップ5511において機器内にデータを留め置きデータへの参照ポインタを返信として送信することを意味する参照送信を指示するジョブスクリプトを発行し、返信された参照ポインタの情報を用いて、ステップ5512においてデータを取得する。ステップ5507において、データの送信宛先がジョブを発行するClient以外の場合は、ステップ5513において機器情報データ内の各担当マネージャの属性表の属性値602（サポートしているデータのアップロード方法）をチェックする。ステップ5513のチェックの結果、参照送信をサポートしている場合は、ステップ5514において送信宛先をパラメータとする参照送信を指示するジョブスクリプトを発行し、ステップ5515においてデータの送信宛先に対して多機能周辺機器201からデータが送信されることを伝えることにより、データの取得指示を送る。ステップ5513のチェックの結果、参照送信をサポートしていない場合は、ステップ5516においてデータ送信を指示するジョブスクリプトを発行し、ステップ5517においてデータを受信し、このデータを送信宛先に再度送信する。ステップ5506のチェックの結果、送信宛先が機器内部である場合は、ステップ5518において、機器情報データ内の各担当マネージャの属性表の属性値602（サポートしているデータのアップロード方法）をチェックする。ステップ5518のチェックの結果、参照送信をサポートしている場合は、ステ

ップ5519において機器内の宛先をパラメータとする参照送信を指示するジョブスクリプトを発光する。ステップ5518のチェックの結果、参照送信をサポートしていない場合は、ステップ5520においてデータ送信を指示するジョブを発行し、ステップ5521においてデータをClientが受信した後に再度データを機器に送り返す。ジョブスクリプトを受信した多機能周辺機器201の各マネージャは、図53、図54、図56（スキャンジョブマネージャ）または図53、図54、図59（フォントマネージャ、フォームオーバーレイマネージャ、ログマネージャ、カラープロファイルマネージャ）に示される処理を行い、指示されたジョブの処理を行う。

【０１３９】（コピージョブの発行）図74はコピージョブの発行を行う処理を表す。コピージョブ発行の指示として、ジョブスクリプトの中に設定する複数の属性値が指定される。ステップ5601において、機器情報データベース502にアクセスして機器情報を取得する。ステップ5602において、コピーマネージャの有する属性表を参照し、指定された属性値が属性表で設定されている各属性値の範囲内にあるかどうかをチェックする。チェックの結果、１つでも範囲内に無い場合は、ステップ5605においてユーザインターフェース上にエラーダイアログを表示するなど、ジョブ発行拒否の処理を行い、終了する。ステップ5602のチェックの結果、全ての属性がコピーマネージャの持つ属性表内の各属性値の範囲内にある場合は、ステップ5603においてコピーマネージャの属性表中に禁止設定属性（属性ID801から805）が存在するかどうかチェックする。コピーマネージャの属性表が禁止設定属性を持つ場合は、ステップ5604においてマネージャが属性表内に持つ各禁止設定属性毎に指定されたジョブの属性が禁止されている設定値の組み合わせとなっていないかどうかチェックする。チェックの結果、禁止されている属性の設定値の組み合わせがある場合は、ステップ5605においてジョブ発行拒否の処理を行い終了する。ステップ5603のチェックの結果、マネージャが禁止設定属性を持たない場合およびステップ5604のチェックの結果ジョブの属性が禁止されている設定値の組み合わせとなっていない場合、ステップ5606においてジョブスクリプトを生成し、発行して終了する。ジョブスクリプトを受信した多機能周辺機器201のコピージョブマネージャは、図53、図54、図57に示される処理を行い、指示されたジョブの管理を行う。

【０１４０】（ジョブ管理コマンドの発行）図75は、ジョブ管理ユーティリティ512が、ジョブ管理コマンドを発行する場合の処理フローを表す。ステップ5701において、対象とするマネージャのサービスIDをパラメータとしてマネージャが保持するジョブ表（図20）をMFP201から取得する。ステップ5702において、取得したジョブ表に含まれるジョブリストの中から適当な方法で管理対象とするジョブの持つジョブIDを選択する。選択手段としては、ユーザインターフェース501を介してディスプレイ6003にジョブのリストを表示し、ユーザに選択させる方法などが含まれる。ステップ5703において、指定されたジョブIDをパラメータとするジョブ管理用コマンドを生成し、発行し終了する。ジョブ管理コマンドを受信した多機能周辺機器201のSupervisorは、図47、図48、図49、図50、図51、図60、図92から図95に示される処理を行い、指示されたジョブの管理を行う。具体的には図75のステップ5703のジョブ管理コマンドがそれぞれ“CancelJob”、“Promote　Job”,” Change Exec Time”, ”Change Expire Time”であった場合にはそのおのおのに対応するジョブマネジャでの行われるジョブ管理処理は図92のジョブ削除処理、図93のジョブ優先度変更処理、図94の実行時刻変更処理、図95の残存限度時刻変更処理である。

【０１４１】（割込みジョブの発行）図96はプリンタドライバ505が、割込みプリントジョブを発行する際にディスプレイ6003に表示するダイアログを示している。ジョブの投入に際し、ユーザーが割込みプリントを指示した場合には割込みオプションを設定させるダイアログ5000がクライアントPCの画面上に現れる。このダイアログでは、割込みプリントのオプションとして何を優先して割込ませるかを設定する。すなわち、多機能周辺機器201は複数のプリントエンジンを有しており、スピード、解像度、フィニッシィング機能等の違いによっていくつかの選択が可能である。

【０１４２】例えば、プリント終了までの待ち時間を最短にしたいのか、あるいは画質は妥協できないので出力時間は少々かかっても画質がよいプリンタを選択したいなどのユーザの好みが設定できる。ここで指定された、ユーザーの好みを第一優先で満足するプリンタを選択し（複数選択することもありえる）それらの実行中のジョブに割込むように指示を出す。図96の5005から5007においてユーザの好みのオプションを一つ選択する。5004には割込みジョブを実行可能なプリントエンジンの候補がリストアップされている。もし、5003の設定においてユーザがプリントエンジン指定5007を選択した場合には、5004のリストアップされたプリントエンジンから好みのものを選択できる。図96の例ではユーザは終了時間優先5005を選択したので、リストアップされているプリンタの全てが使われて、ジョブを分散出力することになる。設定が終了したら5001のプリント実行ボタンを押下して割込みジョブを発行する。

【０１４３】多機能周辺機器201から発行した割込みジョブの終了通知イベントを契機として図９７のジョブ終了ダイアログがユーザPCに表示される。このダイアログには5022に示すユーザの割込み指定と、5023に示す使用されたプリントエンジン名と出力したページ数がリスト表示される。この例では、ユーザは終了時間優先で割込みプリントを行ったので、その時点でジョブ処理可能で

あった3つのプリントエンジンに終了時間が最短になるように負荷分散を行った。ユーザはこのリストに従って出力されたドキュメントを回収することになる。

【０１４４】また、図98のように、ユーザがプリンタ指定のオプションを選択すると、6004に割込みジョブを実行可能な割込みプリントジョブマネージャに関する情報（割込みプリントジョブマネージャの属性の属性値）がリストアップされても良い。なお、各割込みプリントジョブマネージャには仮想的な（論理的な）プリンタ名が付けられていて、括弧の中にはそのプリンタの性能・機能が列挙される。ここで、割込みプリントジョブマネージャの属性の属性値は、図72のステップ5401で取得される。

【０１４５】ユーザが図97のプリンタエンジンのいずれか或い図98の仮想的なプリンタのいずれかを選択すると、それぞれに対応する割込みプリントマネージャのサービスIDをキーとして、機器データベース502を検索し、得られたサブアドレスに割込みプリントジョブのジョブスクリプトを送信する。

【０１４６】また、図99のような画面で、割込みプリントを指定できるようにしても良い。図99の画面では、図66のステップ4806で取得されたサービスIDのリストに含まれているプリントマネージャ、割込みプリントマネージャの一覧が表示されている。なお、こきでは、プリントマネージャ、割込みプリントマネージャについてマネージャ名称がつけられており、そのマネージャ名称が一覧表示されている。ユーザは、割込みプリントを行いたいときには、マネージャの一覧の中から割込みプリントマネージャを選択する。すると、その割込みプリントマネージャのサービスIDをキーとして、機器データベース502を検索し、得られたサブアドレスに割込みプリントジョブのジョブスクリプトを送信する。

【０１４７】画面上では、例えば、プリントマネージャ或いは割込みプリントマネージャの印刷方式、用紙サイズ、カラー印刷の可否、両面印刷の可否、可能な拡大／縮小の範囲などが、ステップ4809で取得された属性値をもとに表示される。ユーザは、この属性値を見て、どのような印刷方式で、どのような用紙サイズで割込みプリントができるか、或いは、カラー印刷の割込みプリントができるか、或いは、両面印刷の割込みプリントができるかを知ることができる。

【０１４８】ここで、印刷方式は、プリントジョブマネージャ或いは割込みプリントマネージャの属性ID2001（jobを実行するController IDのリスト）の属性値からどのプリントコントローラが使用されているかで求められる。また、カラー印刷の可否は、プリントジョブマネージャ或いは割込みプリントマネージャの属性ID1102の属性値から求められる。更に詳細な情報が属性値をもとに表示されることが可能である。

【０１４９】なお、本発明は、単体で存在するコピー機、プリンタ、スキャナ等から構成されるシステムに適用しても、一つの機器からなる装置（例えば、複写機，ファクシミリ装置など）に適用してもよい。また、本発明の目的は、前述した実施形態の機能を実現するソフトウエア（制御プログラム）のプログラムコードを記録した記憶媒体（図76、図77）を、図2に示すようにシステムに供給し、そのシステムの装置（CPU301やCPU6001）が記憶媒体に格納されたプログラムコードを読出し実行することによって達成される。Client PCに図77に示すプログラムやデータを供給する方法として図78に示すようにフロッピーディスクFD7000に記憶させてPC本体7001に供給する方法も一般的である。この場合、記憶媒体から読出されたプログラムコード自体が前述した実施形態の機能を実現することになり、そのプログラムコードを記憶した記憶媒体は本発明を構成することになる。

【０１５０】プログラムコードを供給するための記憶媒体としては、例えば、フロッピディスクやハードディスク以外にも，光ディスク，光磁気ディスク，ＣＤ－ＲＯＭ，ＣＤ－Ｒ，磁気テープ，不揮発性のメモリカード，ＲＯＭなどを用いることができる。また、コンピュータが読出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているＯＳ（オペレーティングシステム）などが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。さらに、記憶媒体から読出されたプログラムコードが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書込まれた後、そのプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるＣＰＵなどが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【０１５１】また、本発明の目的は、前述した実施形態の機能を実現するソフトウエア（制御プログラム）のプログラムコード（図76、図77のプログラムコード）を、プログラム送出装置が図2に示すようなシステムの記憶手段（Memory302、Disk315、Memory6002、Disk6009）に、LANや公衆回線を介して供給し、そのシステムの装置（CPU301やCPU6001）が記憶手段に格納されたプログラムコードを読出し実行することによっても達成される。プログラム送出装置は、主にクライアントサーバ方式のシステムにおけるサーバであり、例えば、webサーバやftpサーバなどが挙げられる。この場合、プログラム送出装置が送出したプログラムコード自体が前述した実施形態の機能を実現することになり、そのプログラムコードを送出した送出装置は本発明を構成することになる。

【０１５２】図79は図1のレーザービームプリンタエン

ジン103,104に適用可能なレーザビームプリンタ（以下、ＬＢＰと略す）の内部構造を示す断面図で、このＬＢＰは、文字パターンデータ等を入力して記録紙に印刷することができる。図79において、8012はＬＢＰ本体であり、供給される文字パターン等を基に、記録媒体である記録紙上に像を形成する。8000は操作のためのスイッチ及びＬＥＤ表示器などが配されている操作パネル、8001はLBP8012全体の制御及び文字パターン情報等を解析するプリンタ制御ユニツトである。このプリンタ制御ユニツト8001は主に文字パターン情報をビデオ信号に変換してレーザドライバ8002に出力する。レーザドライバ8002は半導体レーザ8003を駆動するための回路であり、入力されたビデオ信号に応じて半導体レーザ8003から発射されるレーザ光8004をオン・オフ切替えする。レーザ光8004は回転多面鏡8005で左右方向に振られて静電ドラム8006上を走査する。これにより、静電ドラム8006上には文字パターンの静電潜像が形成される。この潜像は静電ドラム8006周囲の現像ユニツト8007により現像された後、記録紙に転写される。この記録紙にはカットシートを用い、カットシート記録紙はLBP8012に装着した複数種の用紙に対応した複数の用紙カセツト8008に収納され、給紙ローラ8009及び搬送ローラ8010と8011とにより装置内に取込まれて、静電ドラム8006に供給される。

【０１５３】図80は図1のインクジェットプリンタエンジン105に適用可能なインクジェット記録装置ＩＪＲＡの概観図である。同図において、駆動モータ9011の正逆回転に連動して駆動力伝達ギア9010，9008を介して回転するリードスクリュー9004の螺旋溝9003に対して係合するキャリッジＨＣはピン（不図示）を有し、矢印ａ，ｂ方向に往復移動される。このキャリッジＨＣには、インクジェットカートリッジＩＪＣが搭載されている。9001は紙押え板であり、キャリッジの移動方向に亙って紙をプラテン9000に対して押圧する。9006，9007はフォトカプラで、キャリッジのレバー9005のこの域での存在を確認して、モータ9011の回転方向切り換え等を行うためのホームポジション検知手段である。9013は記録ヘッドの前面をキャップするキャップ部材9019を支持する部材で、9012はこのキャップ内を吸引する吸引手段で、キャップ内開口9020を介して記録ヘッドの吸引回復を行う。9014はクリーニングブレードで、9016はこのブレードを前後方向に移動可能にする部材であり、本体支持板9015にこれらが支持されている。ブレードは、この形態でなく周知のクリーニングブレードが本例に適用できることは言うまでもない。又、9018は、吸引回復の吸引を開始するためのレバーで、キャリッジと係合するカム9017の移動に伴って移動し、駆動モータからの駆動力がクラッチ切り換え等の公知の伝達手段で移動制御される。これらのキャッピング、クリーニング、吸引回復は、キャリッジがホームポジション側の領域に来た時にリードスクリュー9004の作用によってそれらの対応位置で所望の処理が行えるように構成されているが、周知のタイミングで所望の作動を行うようにすれば、本例にはいずれも適用できる。

【０１５４】以上説明したように本実施形態によれば、情報処理装置からジョブが投入される論理デバイスの機能を保持し当該論理デバイスに投入されたジョブを管理する論理デバイス制御プログラム（スキャンジョブマネージャー419、プリントジョブマネージャー501から509、コピージョブマネージャー601から608）と周辺機器のデバイスエンジンの機能を保持し当該デバイスエンジンでのジョブを管理する物理デバイス制御プログラム（スキャナコントローラー420、LBPコントローラー510,511、Ink Jetコントローラー512）を用いてジョブを解析することができる。

【０１５５】ここで、物理デバイス制御プログラムは、周辺機器のスキャナエンジンを制御するスキャナ制御プログラム（スキャナコントローラ420）である。物理デバイス制御プログラムは、周辺機器のレーザービームプリンタエンジンを制御するレーザービームプリンタ制御プログラム（LBPコントローラー510,511）である。物理デバイス制御プログラムは、周辺機器のインクジェットプリンタエンジンを制御するインクジェットプリンタ制御プログラム（Ink Jetプリンタコントローラー512）である。また、論理デバイス制御プログラムは、周辺機器のレーザービームプリンタ制御プログラム或いはインクジェットプリンタ制御プログラム或いはレーザービームプリンタ制御プログラム及びインクジェットプリンタ制御プログラムを制御するプリントジョブ制御プログラム（プリントジョブマネージャー501から509）である。また、論理デバイス制御プログラムは、周辺機器のスキャナ制御プログラムを制御するスキャナジョブ制御プログラム（スキャンジョブマネージャー419）である。また、論理デバイス制御プログラムは、周辺機器のスキャナ制御プログラムとレーザービームプリンタ制御プログラム或いはインクジェットプリンタ制御プログラム或いはレーザービームプリンタ制御プログラム及びインクジェットプリンタ制御プログラムを制御するコピージョブ制御プログラム（コピージョブマネージャー601から608）である。

【０１５６】このような構成により、論理デバイス制御プログラムは論理デバイスがジョブの実行のために使用する少なくとも一つのデバイスエンジンとの関係を保持することができる。また、論理デバイスに投入されたジョブに基づいて実際にジョブ実行のために使用するデバイスエンジンを決定することができる。また、論理デバイス制御プログラムが保持する機能を変更できる。論理デバイスに投入されたジョブを実際に実行する物理デバイスで管理するジョブと関係付けて管理することができる。また、論理デバイス制御プログラムは複数存在し、複数の論理デバイス制御プログラムの負荷状態を比較し

て最適な論理デバイス制御プログラムを選択することができる。また、複数の論理デバイス制御プログラムとこれら複数の論理デバイス制御プログラムを統括する統括プログラム（Supervisor410）を有し、当該統括プログラムはそれぞれの前記論理デバイス制御プログラムにジョブを投入する際に使用するサブアドレスと接続インタフェースの種類をそれぞれの論理デバイス制御プログラムに対応して付けて保持することができる。また、統括プログラムはサブアドレスが有効或いは無効かを示す情報の一覧を保持し、有効でないサブアドレスにジョブが投入された場合に当該ジョブを破棄することができる。また、統括プログラムは使用可能なユーザーの認証情報の一覧を保持し、入力したジョブに含まれるユーザー情報が当該認証情報の一覧に含まれない場合に当該ジョブのエラー報知をすることができる。また、統括プログラムは使用可能なユーザーの認証判断処理の種類を示すセキュリティレベルの一覧を保持することができる。また、論理デバイス制御プログラムはジョブを解析中にイベントが発生した場合にイベントデータを情報処理装置へ送信するための接続インタフェースと送信先アドレスの一覧を保持することができる。また、論理デバイス制御プログラムでジョブを解析中にイベントが発生した場合に、情報処理装置へ送信するための接続インタフェースと送信先アドレスの一覧を参照してイベントデータを送信することができる。

【０１５７】また、以上説明したように本実施形態によれば、周辺機器から取得した機能（属性値）に応じて、周辺機器を制御するための制御プログラムのユーザーインタフェースを自動構成（ステップ4813）することができる。また、取得した機能に応じて周辺機器のためのユーザーインタフェースの表示（図81、図82）を制御することができる。また、前記周辺機器の機能の設定範囲に関する情報（属性値）を取得することができる。また、設定範囲に関する情報はジョブ設定が禁止されている属性の組み合わせ（属性ID801からID805）で表現されている。また、周辺機器の機能選択肢に関する情報（属性値）を取得することができる。周辺機器から当該周辺機器の機能を示す属性リスト（属性表）を取得して、当該属性リストの属性IDを指定することによって属性値を取得することができる。また、周辺機器の物理デバイス制御プログラム、論理デバイス制御プログラム、リソース制御プログラムとそれらを統括する統括制御プログラムのそれぞれの機能を示す属性リスト（属性表）を当該周辺機器から取得することができる。また、物理デバイス制御プログラムは周辺機器のスキャナエンジンを制御するスキャナ制御プログラムである。また、物理デバイス制御プログラムは周辺機器のレーザービームプリンタエンジンを制御するレーザービームプリンタ制御プログラムである。

【０１５８】また、物理デバイス制御プログラムは周辺機器のインクジェットプリンタエンジンを制御するインクジェットプリンタ制御プログラムである。また、論理デバイス制御プログラムは周辺機器のレーザービームプリンタ制御プログラム或いはインクジェットプリンタ制御プログラム或いはレーザービームプリンタ制御プログラム及びインクジェットプリンタ制御プログラムを制御するプリントジョブ制御プログラムである。また、論理デバイス制御プログラムは周辺機器のスキャナ制御プログラムを制御するスキャナジョブ制御プログラムである。また、論理デバイス制御プログラムは周辺機器のスキャナ制御プログラムとレーザービームプリンタ制御プログラム或いはインクジェットプリンタ制御プログラム或いはレーザービームプリンタ制御プログラム及びインクジェットプリンタ制御プログラムを制御するコピージョブ制御プログラムである。また、リソース制御プログラムは周辺機器のフォントを管理するフォント制御プログラムである。また、リソース制御プログラムは周辺機器のフォームオーバーレイを管理するフォームオーバーレイ制御プログラムである。また、リソース制御プログラムは周辺機器のログを管理するログ制御プログラムである。また、リソース制御プログラムは周辺機器のカラープロファイルを管理するカラープロファイル制御プログラムである。

【０１５９】また、以上説明したように本実施形態によれば、周辺機器（MFP201）から取得した機能に基づいて前記周辺機器へジョブスクリプトを発行できるか否か判断（ステップ5402から5404、ステップ5502から5504、ステップ5602から5604）することにより、その判断結果に応じてジョブ発行処理を制御できる。

【０１６０】ここで、周辺機器の物理デバイス制御プログラム、論理デバイス制御プログラム、リソース制御プログラム（フォントマネージャー413、フォームオーバーレイマネージャー414、ログマネージャー415、カラープロファイルマネージャー416）とそれらを統括する統括プログラムのそれぞれの機能を示す属性リスト（属性表）が当該周辺機器から取得される。また、ジョブ設定が禁止されている属性の組み合わせ（属性ID801からID805）が取得される。また、ダウンロードジョブ或いはプリントジョブが周辺機器へ発行される。また、アップロードジョブ或いはスキャンジョブが周辺機器へ発行される。また、取得した機能のダウンロード方法のサポートを示す属性（属性ID601）とジョブ発行対象のデータの存在位置に応じて、ジョブ発行方法が選択される。また、取得した機能のアップロード方法のサポートを示す属性（属性ID602）とジョブ発行対象のデータの保存位置に応じて、ジョブ発行方法が選択される。また、ジョブ発行対象のデータは、プリントドキュメントデータである。また、ジョブ発行対象のデータは、スキャンドキュメントデータである。また、ジョブ発行対象のデータは、フォントデータである。また、ジョブ発行対象のデ

ータは、フォームオーバーレイデータである。また、ジョブ発行対象のデータは、カラープロファイルデータである。また、ジョブ発行対象のデータは、ログデータである。また、ジョブ発行対象のデータは、フォントデータである。また、周辺機器から受信したイベントに応じたイベントフォーマットデータを取得し、当該イベントフォーマットデータに基づいて受信したイベントが解析される。また、周辺機器のログデータと当該ログデータのログフォーマットデータを取得し、当該ログフォーマットデータに基づいて前記ログデータが解析される。また、周辺機器の課金データ（カウントデータ）と当該課金データのフォーマットデータを取得し、当該課金フォーマットデータに基づいて前記課金データが解析される。

【０１６１】

【発明の効果】以上述べたように本発明によれば、情報処理装置と接続された周辺機器で、周辺機器の複数のデバイスエンジンの機能を保持し当該デバイスエンジンでのジョブを管理する複数のデバイス制御プログラムの一つを実行してジョブを処理している間に、割り込みジョブが情報処理装置から入力されると、実行しているデバイス制御プログラムとは異なる他のデバイス制御プログラムを選択して実行することができる。

【０１６２】また、デバイス制御プログラムは、割り込みジョブが入力された際に実行していたジョブの実行コンテキストを退避し、該割り込みジョブの実行が終了すると、退避したジョブの実行コンテキストを復帰することができる。

【０１６３】また、割り込みジョブ実行中は、他の割り込みジョブを多重に入力させないようにすることができる。

【０１６４】また、割り込みコピージョブが入力された場合には、周辺機器のスキャナエンジンの機能を保持し当該スキャンエンジンでのジョブを管理するスキャナ制御プログラムにジョブが投入された後に、周辺機器のプリンタエンジンの機能を保持し当該プリンタエンジンでのジョブを管理する任意のプリンタ制御プログラムを選択することができる。

【０１６５】また、情報処理装置のディスプレイには、周辺機器のデバイス制御プログラムに対応する識別情報、例えば、アイコンがリスト表示され、ユーザは、その中のある１つを選択することにより割込みジョブを発行することができる。リスト表示されているアイコンの中には、割込みジョブが発行されたときに選択、実行されるデバイス制御プログラムに対応するアイコンが含まれており、ユーザはこのアイコンを選択することにより割込みジョブが発行される。更に、ユーザは、デバイス制御プログラムに関する情報を見ることにより、どうような割込みジョブを発行することができるのかを知ることができる。

【図面の簡単な説明】

【図１】本発明の一実施形態を示す多機能周辺機器（ＭＦＰ）の構成を示す図である。

【図２】本発明の一実施形態を示すシステム構成を示す図である。

【図３】図１に示したＭＦＰのコントローラのハード構成を示す図である。

【図４】図１に示したＭＦＰのコントローラのソフト構成を示す図である。

【図５】図１に示したＭＦＰのコントローラのソフト構成を示す図である。

【図６】図１に示したＭＦＰのコントローラのソフト構成を示す図である。

【図７】図4に示したＳｕｐｅｒｖｉｓｏｒ４１０の属性表を示す図である。

【図８】図７の属性表の属性ＩＤと型ＩＤの意味を示す図である。

【図９】サブアドレス：サービスＩＤ対応表を示す図である。

【図１０】図9の接続形式ＩＤの意味を示す図である。

【図１１】サブアドレス：タスクタイプ対応表を示す図である。

【図１２】図１１のタスクタイプＩＤの意味を示す図である。

【図１３】ユーザ認証表を示す図である。

【図１４】アクセス制御表を示す図である。

【図１５】セキュリティレベルを示す図である。

【図１６】イベント設定表を示す図である。

【図１７】イベントフォーマット表を示す図である。

【図１８】プリントマネージャの属性表を示す図である。

【図１９】図１８に示す属性表の属性ＩＤと型ＩＤの意味を示す図である。

【図２０】ジョブ表を示す図である。

【図２１】ジョブ依頼表を示す図である。

【図２２】スキャンジョブマネージャの属性表を示す図である。

【図２３】図２２に示す属性表の属性ＩＤと型ＩＤの意味を示す図である。

【図２４】コピージョブマネージャの属性表を示す図である。

【図２５】図２４に示す属性表の属性ＩＤと型ＩＤの意味を示す図である。

【図２６】フォントマネージャの属性表を示す図である。

【図２７】図２６に示す属性表の属性ＩＤと型ＩＤの意味を示す図である。

【図２８】フォント表を示す図である。

【図２９】フォームオーバーレイマネージャの属性表を示す図である。

【図３０】図２９に示す属性表の属性ＩＤと型ＩＤの意味を示す図である。

【図３１】フォームオーバーレイ表を示す図である。

【図３２】ログマネージャの属性表を示す図である。

【図３３】図３２に示す属性表の属性ＩＤと型ＩＤの意味を示す図である。

【図３４】ログ表を示す図である。

【図３５】ログデータの内容を示す図である。

【図３６】ログフォーマット表を示す図である。

【図３７】カラープロファイルマネージャの属性表を示す図である。

【図３８】図３７に示す属性表の属性ＩＤと型ＩＤの意味を示す図である。

【図３９】カラープロファイル表を示す図である。

【図４０】プリンターコントローラの属性表を示す図である。

【図４１】図４０に示す属性表の属性ＩＤと型ＩＤの意味を示す図である。

【図４２】ジョブキュー表を示す図である。

【図４３】図４２のステータスを示す図である。

【図４４】スキャナコントローラの属性表を示す図である。

【図４５】図４４に示す属性表の属性ＩＤと型ＩＤの意味を示す図である。

【図４６】コマンドパケットの構造を示す図である。

【図４７】コマンドパケットの処理を示すフローチャートである。

【図４８】属性表のアクセス処理を示すフローチャートである。

【図４９】サービスＩＤ一覧問合わせ処理のフローチャートである。

【図５０】サブアドレス問合わせ処理のフローチャートである。

【図５１】タスクタイプを指定したサービスＩＤの問合わせ処理のフローチャートである。

【図５２】ジョブの構造を示す図である。

【図５３】各マネージャでのジョブスクリプト処理のフローチャートである。

【図５４】各マネージャでのジョブスクリプト処理のフローチャートである。

【図５５】プリントジョブマネージャでのジョブ処理のフローチャートである。

【図５６】スキャンジョブマネージャでのジョブ処理のフローチャートである。

【図５７】コピージョブマネージャでのジョブ処理のフローチャートである。

【図５８】フォントマネージャ、フォームオーバーレイマネージャ、ログマネージャ、カラープロファイルマネージャでのジョブ処理（ダウンロード）のフローチャートである。

【図５９】フォントマネージャ、フォームオーバーレイマネージャ、ログマネージャ、カラープロファイルマネージャでのジョブ処理（アップロード）のフローチャートである。

【図６０】各マネージャでのジョブ管理のフローチャートである。

【図６１】イベント送信のフローチャートである。

【図６２】機器からのデータ（スクリプト）送信のフローチャートである。

【図６３】Ｃｌｉｅｎｔ　ＰＣのハード構成を示す図である。

【図６４】Ｃｌｉｅｎｔ　ＰＣのソフトウエア（制御プログラム）構成を示す図である。

【図６５】パケット生成・送信処理のフローチャートである。

【図６６】Ｃｌｉｅｎｔ側での情報取得およびソフトウエア（制御プログラム）自動構成のフローチャートである。

【図６７】タスクタイプを指定したジョブ発行のフローチャートである。

【図６８】イベントの構造を示す図である。

【図６９】イベント送信のフローチャートである。

【図７０】ログ処理のフローチャートである。

【図７１】課金情報の取得・処理のフローチャートである。

【図７２】ジョブ発行（プリントジョブ、データダウンロード）のフローチャートである。

【図７３】ジョブ発行（スキャンジョブ、データアップロード）のフローチャートである。

【図７４】コピージョブ発行のフローチャートである。

【図７５】ジョブ管理コマンド発行のフローチャートである。

【図７６】ＭＦＰのＤＩＳＫ３１５のメモリマップである。

【図７７】Ｃｌｉｅｎｔ　ＰＣのＤＩＳＫ６００９のメモリマップである。

【図７８】Ｃｌｉｅｎｔ　ＰＣの構成を示す図である。

【図７９】ＭＦＰのレーザービームプリンタエンジンに適用可能なＬＢＰを示す図である。

【図８０】ＭＦＰのインクジェットプリンタエンジンに適用可能なＩＪＲＡを示す図である。

【図８１】ディスプレイに表示される表示画面のサンプルを示す図である。

【図８２】ディスプレイに表示される表示画面のサンプルを示す図である。

【図８３】ジョブスケジューリング表を示す図である。

【図８４】実行時刻順ジョブキュー表を示す図である。

【図８５】ジョブコンテキスト表を示す図である。

【図８６】プリントジョブマネジャでのジョブ処理（2）を示す図である。

【図８７】プリントジョブマネジャでのジョブ処理（３）を示す図である。

【図８８】プリントジョブマネジャでのジョブ処理（４）を示す図である。

【図８９】割込みプリントジョブマネジャでのジョブ処理を示す図である。

【図９０】プリントコントローラのジョブ処理を示す図である。

【図９１】割込みコピージョブマネジャでのジョブ処理を示す図である。

【図９２】各マネージャでのジョブ管理(2)のフローチャートである。

【図９３】各マネージャでのジョブ管理(3)のフローチャートである。

【図９４】各マネージャでのジョブ管理(4)のフローチャートである。

【図９５】各マネージャでのジョブ管理(5)のフローチャートである。

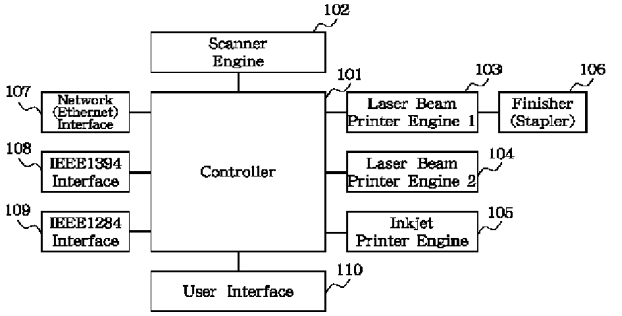
【図96】割込みプリント実行時のオプション設定ダイアログである。

【図97】割込みプリント終了時の通知ダイアログである。

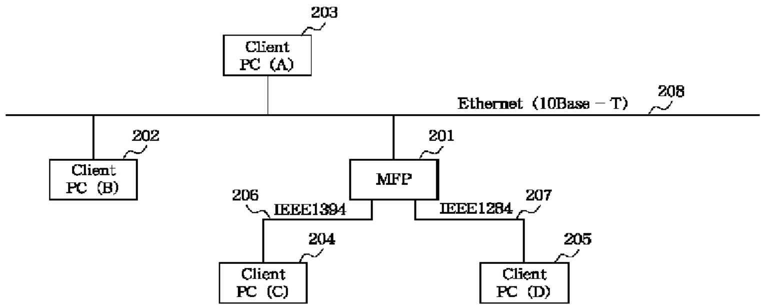
【図９８】割込みプリント実行時のオプション設定ダイアログである。

【図９９】割込みプリント実行時のユーザインターフェースの一例である。

【図１】



【図２】



【図７】

属性表（スーパーバイザー）

701　702　703

属性ID	型ID	値
10	1	0
11	1	0
100	11	10,11,100,101,102,103,104,105,···
101	11	1,2,3,7,9
102	11	1,3
103	1	1
104	11	1001,1002,1003,1004
105	11	1003
301	3	"SuperMFP − 1"
302	13	"Japanese","English"
303	1	"English"
401	11	45,78,34,13
402	11	565,537,545,523
403	11	100,100,100,200
404	13	0.56,0.78,0.34,0.55
405	3	135.45
406	3	500.00
501	11	200,399,432,234
502	52	Table Data
503	53	Table Data
1001	101	Table Data
1002	102	Table Data
1003	103	Table Data
1004	104	Table Data
2001	11	21,22,23,24

【図１４】

アクセス制御表

1101　1102　1103

SID	Security Level	UID
0	1	−
1	1	−
2	2	2355,3254
3	3	3254,4665
102	2	3254

【図１０】

・サービスを受ける時に、どのサブアドレスにジョブを投入したら良いかを記述したもの。

・その接続形態が有効かどうか（使用させるかどうか）もこの表で管理する。

・接続形式ID
　0：Internal
　1：TCP／IP
　2：IEEE1284.4
　3：SBB－2

【図１２】

・Service IDがどのような種類のサービスであるかを表す表
・Task TypeはCPCAのタスクタイプに相当
・この表はDispatcherからも参照され、コマンドを各Managerに配信するためにも使用される
・TaskType
　0：Supervisor
　1：Print
　2：Scan
　3：Copy
　11：Interrupt Print
　13：Interrupt Copy
　101：Font
　102：Form Overlay
　103：Log
　104：Color Profile

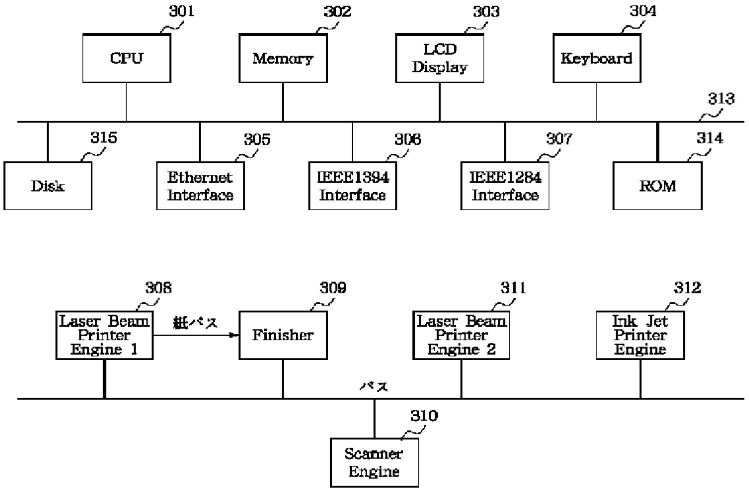
【図１３】

ユーザ認証表

1001　1002　1003

UID	パスワード	管理者
2355	2345	NO
3254	2223	NO
4665	1145	YES
2664	1244	NO

【図３】



【図１５】

・サービス毎のセキュリティレベルを設す
・Security Level
　0：何も区別しない
　1：管理者のみ区別する
　2：認証されたユーザのみ使用可能（認証にUIDのみ使用）
　3：認証されたユーザのみ使用可能　（認証にUIDとパスワードを使用）
・表のUIDの列は、Security Levelが2と3の時のみ使用

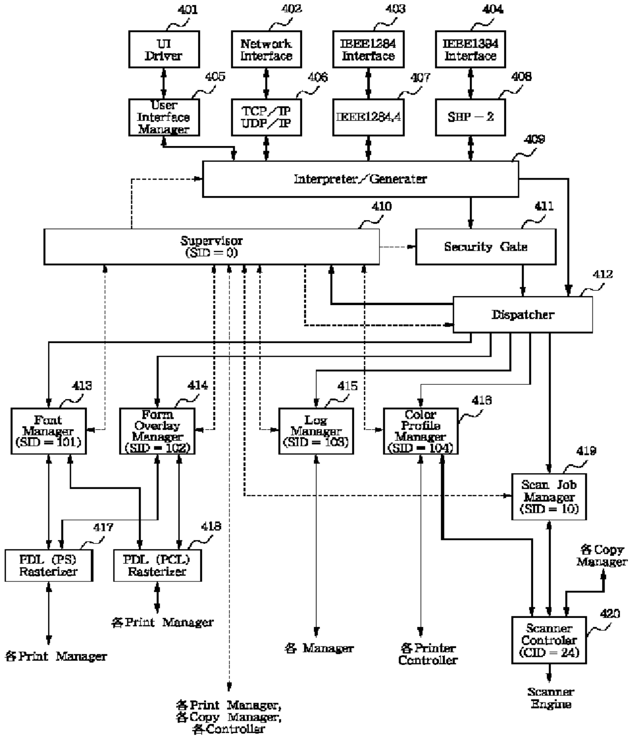
【図１７】

イベントフォーマット表

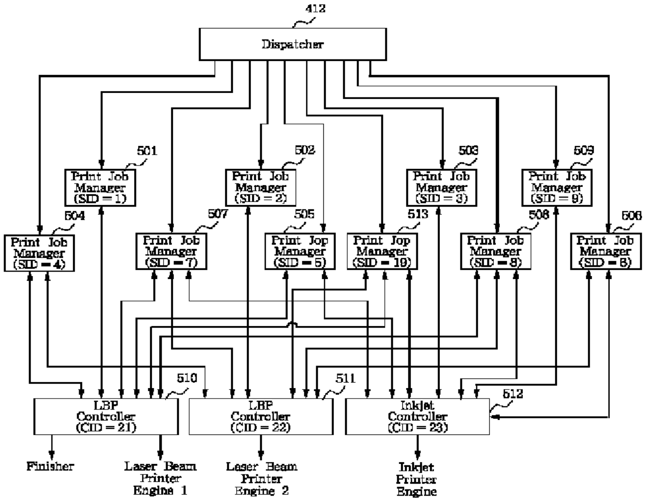
1301　1302

Event ID	イベントフォーマット
200	576,756
399	566
432	698
234	600

【図４】



【図５】


412
Dispatcher
501
Print Job Manager (SID＝1)
502
Print Job Manager (SID＝2)
503
Print Job Manager (SID＝3)
509
Print Job Manager (SID＝9)
504
Print Job Manager (SID＝4)
507
Print Job Manager (SID＝7)
505
Print Jop Manager (SID＝5)
513
Print Jop Manager (SID＝19)
508
Print Job Manager (SID＝8)
506
Print Job Manager (SID＝6)
510
LBP Controller (CID＝21)
511
LBP Controller (CID＝22)
512
Inkjet Controller (CID＝23)
Finisher
Laser Beam Printer Engine 1
Laser Beam Printer Engine 2
Inkjet Printer Engine

【図１６】

イベント設定表（Supervisor）

1201　1202　1203

Event ID	接続形式ID	通知先アドレス
200	0	
399	1	123,222,111,321：9340
432	2	35
234	3	12

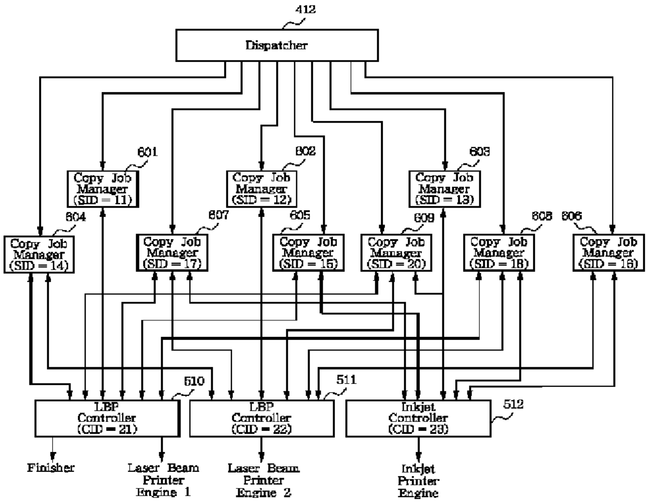
【図２１】

ジョブ依頼表

1503　1504　1505

Job ID	Control ID	Job ID
1	21	100
1	22	101
2	21	102
2	22	403
3	－	－

【図６】


412
Dispatcher
601
Copy Job Manager (SID = 11)
602
Copy Job Manager (SID = 12)
603
Copy Job Manager (SID = 13)
604
Copy Job Manager (SID = 14)
607
Copy Job Manager (SID = 17)
605
Copy Job Manager (SID = 15)
609
Copy Job Manager (SID = 20)
608
Copy Job Manager (SID = 18)
606
Copy Job Manager (SID = 16)
510
LBP Controller (CID = 21)
511
LBP Controller (CID = 22)
512
Inkjet Controller (CID = 23)
Finisher
Laser Beam Printer Engine 1
Laser Beam Printer Engine 2
Inkjet Printer Engine

【図８】

・Supervisor が持つ属性を表す
・属性IDの意味するところはClient は既に知っている。

・属性ID

10	TaskType
11	Service ID
100	属性IDのリスト
101	サポートしているオペレーション
102	サポートしているSecurity Level
103	現在のSecurity Level
104	管理者のみが変更できる属性リスト
105	管理者のみが取得できる属性リスト
301	機種名
302	サポートしている言語
303	現在の言語
401	現在のカウントデータ
402	カウントデータフォーマット
403	カウント制限データ
404	カウント単価データ
405	現在課金データ
406	課金制限データ
501	サポートしているイベントリスト
502	イベント設定表
503	イベントフォーマット表
1001	サブアドレス：SID対応表
1002	SID：TaskType表
1003	ユーザ認証表
1004	アクセス制限表
2001	Controller IDリスト

・型ID

0	Boolean
1	整数
2	実数
3	文字列
11	整数のリスト
12	実数のリスト
13	文字列のリスト
52	イベント設定表形式
53	イベントフォーマット表形式
101	サブアドレス：SID表形式
102	SID：TaskType表形式
103	ユーザ認証表形式
104	アクセス制御表形式

【図１１】

サービスID：タスクタイプ対応表

901　902

Service ID	Task Type
0	0
1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	2
11	3
12	3
13	3
14	3
15	3
16	3
17	3
18	3
19	11
20	13
101	101
102	102
103	103
104	104
21	201
22	201
23	201
24	202

【図９】

サブアドレス：サービスID対応表

801　802　803　804　805　806

接続形式ID	サブアドレス	Service ID	有効フラグ	有効ユーザ	無効ユーザ
0	0	0	YES		
0	1	1	YES	2355,5678	
0	2	2	YES		1234,2345
1	9600	0	YES		
1	9601	1	YES		
1	9602	2	YES	1234,2345	
1	8001	101	YES	1234	
1	8002	102	YES		
2	45	0	YES		
2	46	1	YES		
2	47	2	YES		
2	68	8	NO		
2	89	10	YES		
3	23	0	YES		
3	52	1	YES		
3	98	17	YES		

【図１８】

属性表（プリントジョブマネージャ）

1401　1402　1403

属性ID	型ID	値
10	1	1
11	1	4
100	11	10,11,100,101,104,401,…
101	11	11,14,15,17
104	11	1104,1105
401	11	45,78,34,13
402	11	565,537,545,523
403	11	100,100,100,200
404	13	0.56,0.78,0.34,0.55
405	3	135.45
406	3	500.00
501	11	200,399,432,234
502	52	Table Data
601	11	1,2
801	81	4354:3254, 2464, 2456, 4356:124, 535, 6
802	82	3254:1:4324:43143, 2342, 6543, 3245
803	83	5425:2:25432:34324:2:12453, 3424, 2343
804	84	5425:2:25432:4:5425:5:655:2435, 563
805	85	22414:3:4324231:5:43243:2:32144:455
1101	11	3,5
1102	0	NO
1103	11	1,2
1104	1	400
1105	1	100
2001	11	21,22
2002	0	NO
2003	203	Table Data
2004	204	Table Data
2005	205	Table Data
2006	0	YES

【図３４】

Log表

2301　2302

Log ID	Log Data File名
1	1700
2	1701
3	1702

【図２０】

ジョブ表

1501　1502　1506　1507　1508

Job ID	Job File Name	実行優先度	指定実行時刻	残存限度時刻
1	P0001	3	13：45	14：45
2	P1002	3	13：45	14：45
3	P2001	1	−	−

【図１９】

・Print　Managerが持つ属性を表す
・属性IDの意味するところはClientは既に知っている。

・属性ID

10	TaskType
11	Service　ID
100	属性IDのリスト
101	Jobスクリプトでサポートしているオペレーション
104	管理者のみが変更できる属性リスト
401	現在のカウントデータ
402	カウントデータフォーマット
403	カウント制限データ
404	カウント単位データ
405	現在の課金データ
406	課金制限データ
501	サポートしているイベントリスト
502	イベント設定表
601	サポートしているデータのダウンロード方法
801	設定が禁止されている属性の組み合わせリスト（1型）
802	設定が禁止されている属性の組み合わせリスト（2型）
803	設定が禁止されている属性の組み合わせリスト（3型）
804	設定が禁止されている属性の組み合わせリスト（4型）
805	設定が禁止されている属性の組み合わせリスト（6型）
1101	サポートしているPDLのリスト
1102	カラー印刷可能か
1103	サポートしているFinishingの種類
1104	設定可能最高解像度
1105	設定可能最低解像度
2001	Jobを実行するController IDリスト
2002	Controler自動選択可能かどうか
2003	Job表
2004	Job依頼表
2005	Jobスケジュール表
2006	ジョブ管理設定可能かどうか

・型ID

0	Boolean
1	整数
2	実数
3	文字列
11	整数のリスト
12	実数のリスト
13	文字列のリスト
52	イベント設定表形式
81	禁止属性組み合わせリスト１形式
82	禁止属性組み合わせリスト２形式
83	禁止属性組み合わせリスト３形式
203	ジョブ表形式
204	ジョブ依頼表形式
205	ジョブスケジュール表形式

【図２２】

属性表（スキャンジョブマネージャ）

1601　1602　1603

属性ID	型ID	値
10	1	2
11	1	10
100	11	10,11,100,101,104,401,…
101	11	21,22,25,29
104	11	602
401	11	45,78,34,13
402	11	565,537,545,523
403	11	100,100,100,200
404	13	0.56,0.78,0.34,0.55
405	3	135.45
406	3	500.00
501	11	200,399,432,234
502	52	Table　Data
602	11	1,2
1201	11	3,5
1202	0	YES
1203	11	100,200,300,400
2001	11	24
2002	0	NO
2003	203	Table　Data

【図２３】

・Scan　Managerが持つ属性を表す
・属性IDの意味するところはClientは既に知っている。

・属性ID

10	TaskType
11	Service　ID
100	属性IDのリスト
101	Jobスクリプトでサポートしているオペレーション
104	管理者のみが変更できる属性リスト
401	現在のカウントデータ
402	カウントデータフォーマット
403	カウント制限データ
404	カウント単価データ
405	現在の課金データ
406	課金制限データ
501	サポートしているイベントリスト
502	イベント設定表
602	サポートしているデータのアップロード方法
1201	サポートしているImage　Formatのリスト
1202	カラースキャン可能か
1203	設定可能な解像度のリスト
2001	Jobを実行するController IDリスト
2002	Controller自動選択可能かどうか
2003	Job表

・型ID

0	Boolean
1	整数
2	実数
3	文字列
11	整数のリスト
12	実数のリスト
13	文字列のリスト
52	イベント設定表形式
203	Job表形式

【図２４】

属性表（コピージョブマネージャ）

1701　1702　1703

属性ID	型ID	値
10	1	3
11	1	18
100	11	10,11,100,101,104,401,…
101	11	31,32,35
104	11	1304,1305
401	11	45,78,34,13
402	11	565,537,545,523
403	11	100,100,100,200
404	13	0.56,0.78,0.34,0.55
405	3	135.45
406	3	500.00
501	11	200,399,432,234
502	52	Table　Data
1302	0	NO
1303	11	1,2
1304	1	400
1305	1	100
2001	11	21,22,23,24
2002	0	YES
2003	203	Table　Data
2004	204	Table　Data
2005	203	Table　Data
2006	0	YES

【図２５】

・Copy　Managerが持つ属性を表す
・属性IDの意味するところはClientは既に知っている。

・属性ID

10	TaskType
11	Service　ID
100	属性IDのリスト
101	Jobスクリプトでサポートしているオペレーション
104	管理者のみが変更できる属性リスト
401	現在のカウントデータ
402	カウントデータフォーマット
403	カウント制限データ
404	カウント単価データ
405	現在の課金データ
406	課金制限データ
501	サポートしているイベントリスト
502	イベント設定表
1302	カラー印刷可能か
1303	サポートしているFinishingの種類
1304	設定可能最高解像度
1305	設定可能最低解像度
2001	Jobを実行するController IDリスト
2002	Controller自動選択可能かどうか
2003	Job表
2004	Job依頼表
2005	Jobスケジュール表
2006	ジョブ管理設定可能かどうか

・型ID

0	Boolean
1	整数
2	実数
3	文字列
11	整数のリスト
12	実数のリスト
13	文字列のリスト
52	イベント設定表形式
203	ジョブ表形式
204	ジョブ依頼表形式
205	ジョブスケジュール表形式

【図２６】

属性表（フォントマネージャ）

1801　1802　1803

属性ID	型ID	値
10	1	101
11	1	101
100	11	10,11,100,101,104,401,…
101	11	110,111,112,
104	11	
401	11	45,78,34,13
402	11	565,537,545,523
403	11	100,100,100,200
404	13	0.56,0.78,0.34,0.55
405	3	135.45
406	3	500.00
501	11	200,399,432,234
502	52	Table　Data
601	11	1,2
602	11	1,2
1501	11	3,5
1502	1	10
1503	1	3
1504	150	Table　Data

【図３５】

ログデータの内容（File名：1700）

2401　2402　2403

Log　Format　ID	Date／Time	値
1	98／1／3　1：23	12345,4,1.23, "OK"
2	98／1／3　1：25	22345,12,4.22, "OK"
1	98／1／3　1：43	98763,11,3.45, "Canceled"
2	98／1／3　2：33	77623,5,1.44, "NG"
1	98／1／3　3：32	12346,5,1.66, "OK"

【図27】

・Font Managerが持つ属性を表す
・属性IDの意味するところはClientは既に知っている。

・属性ID

10	TaskType
11	Service ID
100	属性IDのリスト
101	Resourceスクリプトでサポートしているオペレーション
104	管理者のみが変更できる属性リスト
401	現在のカウントデータ
402	カウントデータフォーマット
403	カウント制限データ
404	カウント単価データ
405	現在の課金データ
406	課金制限データ
501	サポートしているイベントリスト
502	イベント設定表
601	サポートしているデータのダウンロード方法
602	サポートしているデータのアップロード方法
1501	サポートしているFont Typeのリスト
1502	最大保持可能Font数
1503	現在のFont数
1504	保持しているFontのリスト

・型ID

0	Boolean
1	整数
2	実数
3	文字列
11	整数のリスト
12	実数のリスト
13	文字列のリスト
52	イベント設定表形式
150	Font表

【図28】

Font表

1901　1902　1902　1903

Font ID	Font Type	Font名	Font Data File名
1	1	Gothic	1500
2	2	Times	1501
3	1	Helvetica	1502

【図29】

属性表（フォームオーバーレイマネージャ）

2001　2002　2003

属性ID	型ID	値
10	1	102
11	1	102
100	11	10,11,100,101,104,401,…
101	11	110,111,112,
104	11	
401	11	45,78,34,13
402	11	565,537,546,523
403	11	100,100,100,200
404	13	0.56,0.78,0.34,0.55
405	3	135.45
406	3	500.00
501	11	200,399,432,234
502	52	Table Data
601	11	1,2
602	11	1,2
1601	11	1,2
1602	1	10
1603	1	3
1604	160	Table Data

【図30】

・Form Overlay Managerが持つ属性を表す
・属性IDの意味するところはClientは既に知っている。

・属性ID

10	TaskType
11	Service ID
100	属性IDのリスト
101	Resourceスクリプトでサポートしているオペレーション
104	管理者のみが変更できる属性リスト
401	現在のカウントデータ
402	カウントデータフォーマット
403	カウント制限データ
404	カウント単価データ
405	現在の課金データ
406	課金制限データ
501	サポートしているイベントリスト
502	イベント設定表
601	サポートしているデータのダウンロード方法
602	サポートしているデータのアップロード方法
1601	サポートしているフォーマットのリスト
1602	最大保持可能Form数
1603	現在のForm数
1604	保持しているFormのリスト

・型ID

0	Boolean
1	整数
2	実数
3	文字列
11	整数のリスト
12	実数のリスト
13	文字列のリスト
52	イベント設定表形式
160	Form表

【図31】

Form表

2101　2102　2103　2104

Form ID	Format	Form名	Form Data File名
1	1	Confidential	1600
2	2	Reference	1601
3	1	Logo	1602

【図32】

属性表（ログマネージャ）

2201　2202　2203

属性ID	型ID	値
10	1	103
11	1	103
100	11	10,11,100,101,104,401,…
101	11	110,111,112,
104	11	
401	11	45,78,34,13
402	11	565,537,546,523
403	11	100,100,100,200,
404	13	0.56,0.78,0.34,0.55
405	3	135.45
406	3	500.00
501	11	200,399,432,234
502	52	Table Data
602	11	1,2
1703	1	3
1704	170	Table Data
1705	171	Table Data

【図33】

・Log Managerが持つ属性を表す
・属性IDの意味するところはClientは既に知っている。

・属性ID

10	TaskType
11	Service ID
100	属性IDのリスト
101	Resourceスクリプトでサポートしているオペレーション
104	管理者のみが変更できる属性リスト
401	現在のカウントデータ
402	カウントデータフォーマット
403	カウント制限データ
404	カウント単価データ
405	現在の課金データ
406	課金制限データ
501	サポートしているイベントリスト
502	イベント設定表
602	サポートしているデータのアップロード方法
1703	保持されているLog数
1704	保持しているLogのリスト
1705	Logのフォーマット表

・型ID

0	Boolean
1	整数
2	実数
3	文字列
11	整数のリスト
12	実数のリスト
13	文字列のリスト
52	イベント設定表形式
170	Log表
171	Logフォーマット表形式

【図36】

ログフォーマット表

2501　2502

Log Format ID	フォーマット
1	1：701,1：565,1：765,1：777
2	11：701,11：565,11：765,11：777

【図39】

カラープロファイル表

2701　2702　2703

Profile ID	Format	Profile Data File名
1	3	1800
2	5	1801
3	3	1802

【図42】

ジョブキュー表

2901　2902　2903

Job ID	Status	File Name
100	1	C100
101	2	C101
102	3	C102
103	3	C103
104	3	C104
105	4	C105

【図43】

・Printer Controllerで実行中のJobおよび実行待ちのジョブのリスト
・Stausの意味
1：実行終了
2：Enjineで実行中
3：実行待ち
4：実行サスペンド中

【図３７】

属性表（カラープロファイルマネージャ）

2601　2602　2603

属性ID	型ID	値
10	1	104
11	1	104
100	11	10,11,100,101,104,401,…
101	11	110,111,112,
104	11	
401	11	45,78,34,13
402	11	565,537,545,523
403	11	100,100,100,200
404	13	0.56,0.78,0.34,0.55
405	3	135.45
406	3	500.00
501	11	200,399,432,234
502	52	Table Data
601	11	1,2
602	11	1,2
1801	11	3,5
1802	1	10
1803	1	3
1804	180	Table Data

【図３８】

・Color Profile Managerが持つ属性を表す
・属性IDの意味するところはClientは既に知っている。

・属性ID

10	TaskType
11	Service ID
100	属性IDのリスト
101	Resourceスクリプトでサポートしているオペレーション
104	管理者のみが変更できる属性リスト
401	現在のカウントデータ
402	カウントデータフォーマット
403	カウント制限データ
404	カウント単価データ
405	現在の課金データ
406	課金制限データ
501	サポートしているイベントリスト
502	イベント設定表
601	サポートしているデータのダウンロード方法
602	サポートしているデータのアップロード方法
1801	サポートしているフォーマットのリスト
1802	最大保持可能Profile数
1803	現在のProfile数
1804	保持しているProfileのリスト

・型ID

0	Boolean
1	整数
2	実数
3	文字列
11	整数のリスト
12	実数のリスト
13	文字列のリスト
52	イベント設定表形式
180	Profile表

【図４０】

属性表（プリンタコントローラ）

2801　2802　2803

属性ID	型ID	値
12	1	1
13	1	21
100	11	12,13,104…
104	11	
401	11	45,78,34,13
402	11	565,537,545,523
403	11	100,100,100,200
404	13	0.56,0.78,0.34,0.55
405	3	135.45
406	3	500.00
501	11	200,399,432,234
502	52	Table Data
5001	0	NO
5002	11	1,2,3
5003	1	1200
5004	1	100
5005	500	Table Data
5006	0	0
5007	501	Table Data
5008	501	Table Data

【図４１】

・Printer Controllerが持つ属性を表す
・属性IDの意味するところはClientは既に知っている。

・属性ID

12	Controller Type
13	Controller ID
100	属性IDのリスト
104	管理者のみが変更できる属性リスト
401	現在のカウントデータ
402	カウントデータフォーマット
403	カウント制限データ
404	カウント単価データ
405	現在の課金データ
406	課金制限データ
501	サポートしているイベントリスト
502	イベント設定表
5001	カラー印刷可能か
5002	サポートしているFinishingの種類
5003	設定可能最高解像度
5004	設定可能最低解像度
5005	Job Que表
5006	割り込みジョブ実行中か
5007	実行中ジョブコンテキスト表
5008	被割り込みジョブコンテキスト表

・型ID

0	Boolean
1	整数
2	実数
3	文字列
11	整数のリスト
12	実数のリスト
13	文字列のリスト
52	イベント設定表形式
500	Job Que表形式

【図４４】

属性表（スキャナコントローラ）

3001　3002　3003

属性ID	型ID	値
12	1	2
13	1	24
100	11	12,13,104,…
104	11	
401	11	45,78,34,13
402	11	565,537,545,523
403	11	100,100,100,200
404	13	0.56,0.78,0.34,0.55
405	3	135.45
406	3	500.00
501	11	200,399,432,234
502	52	Table Data
6001	0	YES
6002	12	200.00,300.00
6003	1	1200
6004	1	100

【図４５】

・Scanner Controllerが持つ属性を表す
・属性IDの意味するところはClientは既に知っている。

・属性ID

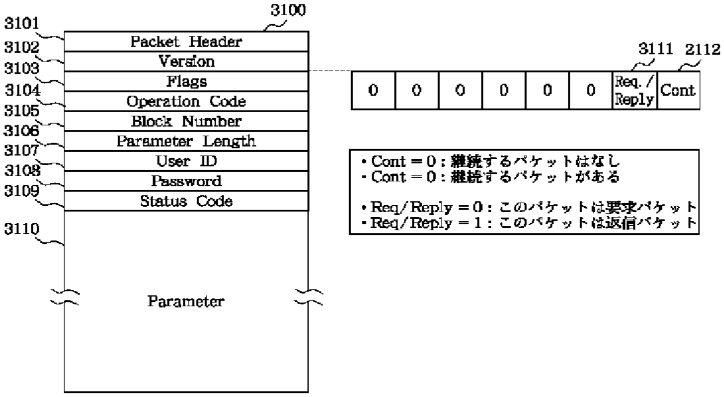
12	Controller Type
13	Controller ID
100	属性IDのリスト
104	管理者のみが変更できる属性リスト
401	現在のカウントデータ
402	カウントデータフォーマット
403	カウント制限データ
404	カウント単価データ
405	現在の課金データ
406	課金制限データ
501	サポートしているイベントリスト
502	イベント設定表
6001	カラースキャン可能か
6002	最大原稿サイズ
6003	設定可能最高解像度
6004	設定可能最低解像度

・型ID

0	Boolean
1	整数
2	実数
3	文字列
11	整数のリスト
12	実数のリスト
13	文字列のリスト
52	イベント設定表形式

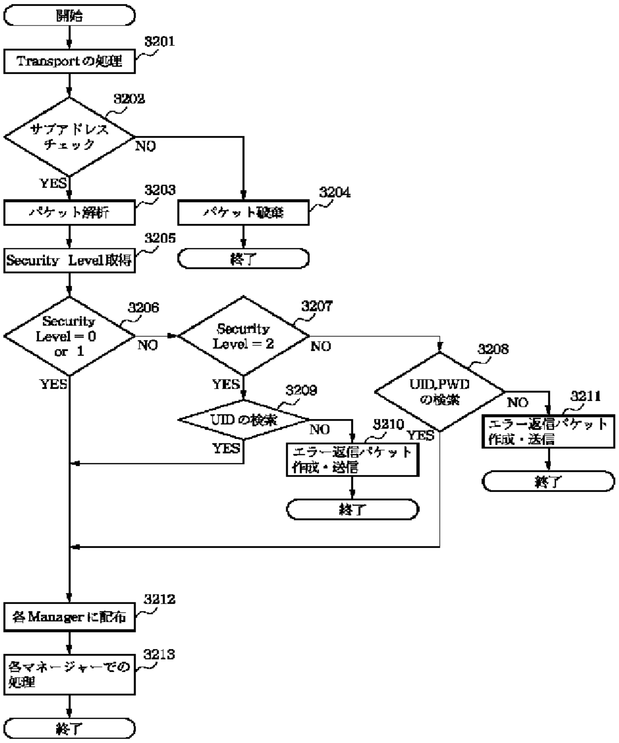
【図４６】

コマンドパケットの構造



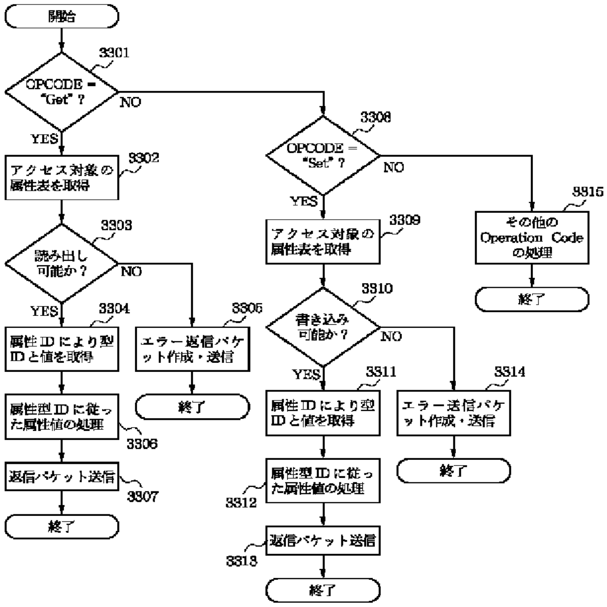
【図４７】

コマンドパケットの処理フロー



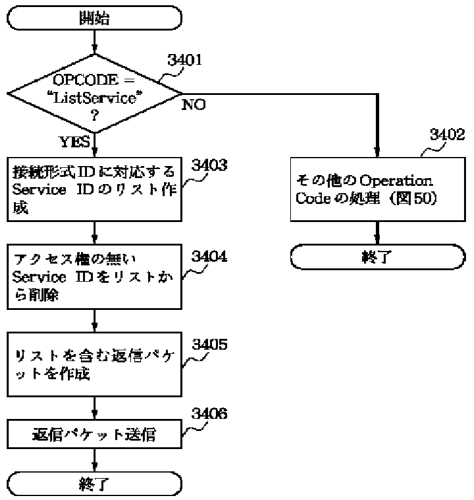
【図４８】

属性表のアクセス処理フロー


開始
3301
OPCODE＝"Get"？
NO
YES
3302
アクセス対象の属性表を取得
3303
読み出し可能か？
NO
YES
3304
属性IDにより型IDと値を取得
3305
エラー返信パケット作成・送信
終了
3306
属性型IDに従った属性値の処理
3307
返信パケット送信
終了
3308
OPCODE＝"Set"？
NO
YES
3309
アクセス対象の属性表を取得
3310
書き込み可能か？
NO
YES
3311
属性IDにより型IDと値を取得
3312
属性型IDに従った属性値の処理
3313
返信パケット送信
終了
3314
エラー送信パケット作成・送信
終了
3315
その他のOperation Codeの処理
終了

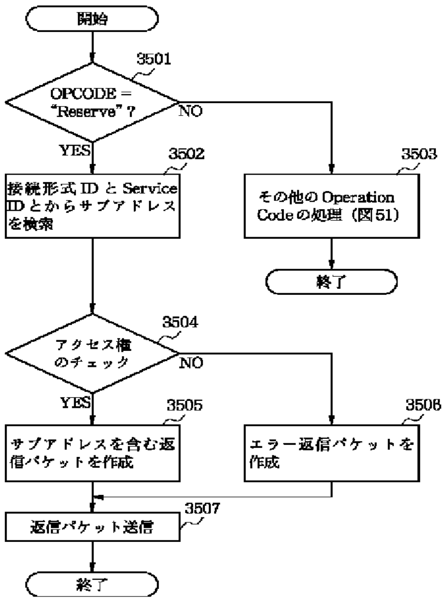
【図４９】

サービスID一覧問合わせ処理


開始
3401
OPCODE = "ListService" ?
NO
YES
3402
その他のOperation Codeの処理（図50）
終了
3403
接続形式IDに対応するService IDのリスト作成
3404
アクセス権の無いService IDをリストから削除
3405
リストを含む返信パケットを作成
3406
返信パケット送信
終了

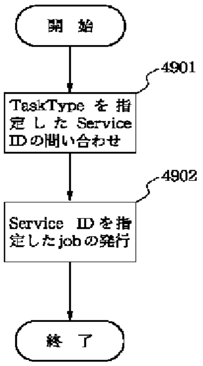
【図５０】

サブアドレス問合わせ処理



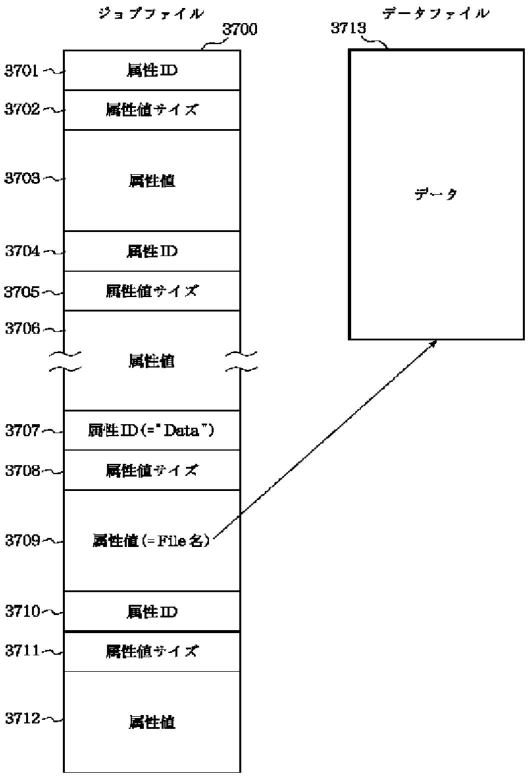
【図６７】

タスクタイプを指定したジョブの発行


開　始
4901
TaskTypeを指定したService IDの問い合わせ
4902
Service IDを指定したjobの発行
終　了

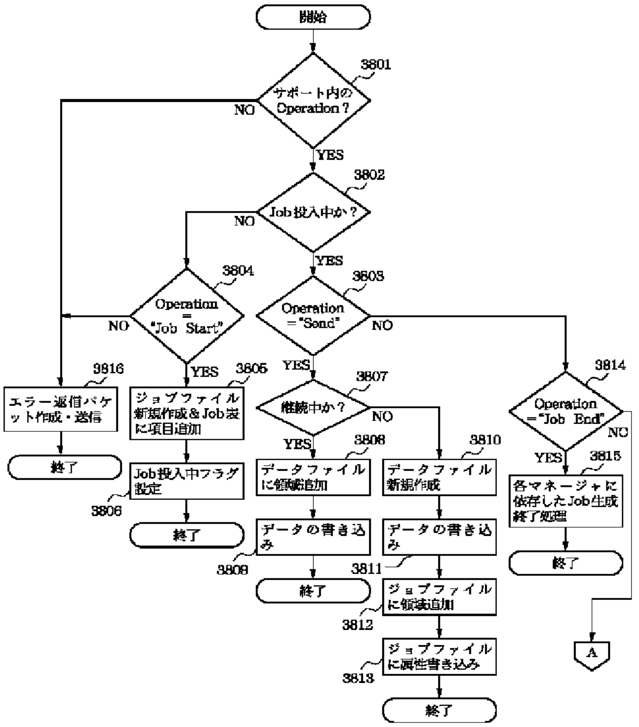
【図５２】

ジョブの構造（データを含む場合）



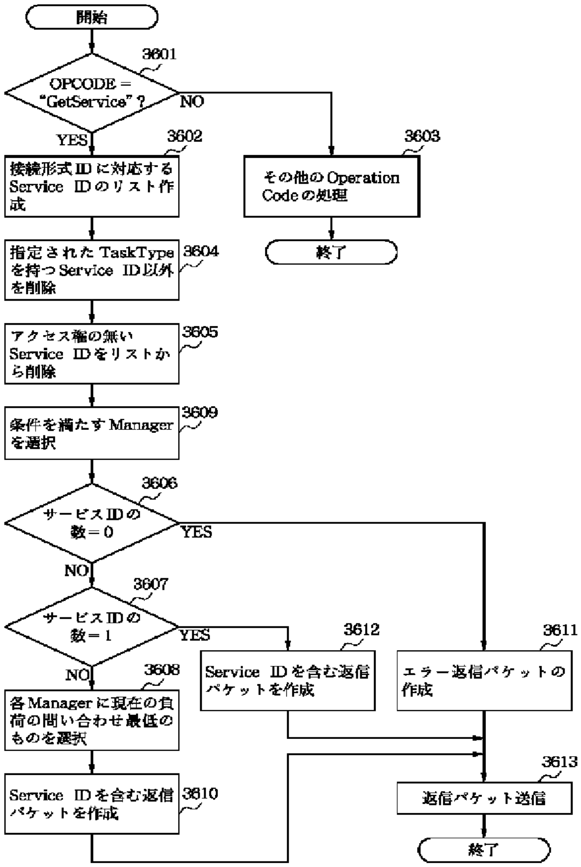
【図５３】

各マネージャでのジョブスクリプト処理



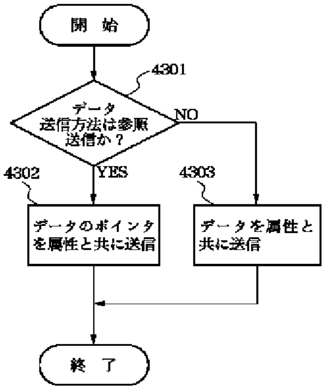
【図51】

タスクタイプを指定したサービスIDの問い合わせ処理


開始
3601
OPCODE =
"GetService"?
NO
YES
3602
接続形式IDに対応するService IDのリスト作成
3603
その他のOperation Codeの処理
終了
3604
指定されたTaskTypeを持つService ID以外を削除
3605
アクセス権の無いService IDをリストから削除
3609
条件を満たすManagerを選択
3606
サービスIDの数＝0
YES
NO
3607
サービスIDの数＝1
YES
NO
3608
各Managerに現在の負荷の問い合わせ最低のものを選択
3612
Service IDを含む返信パケットを作成
3611
エラー返信パケットの作成
3610
Service IDを含む返信パケットを作成
3613
返信パケット送信
終了

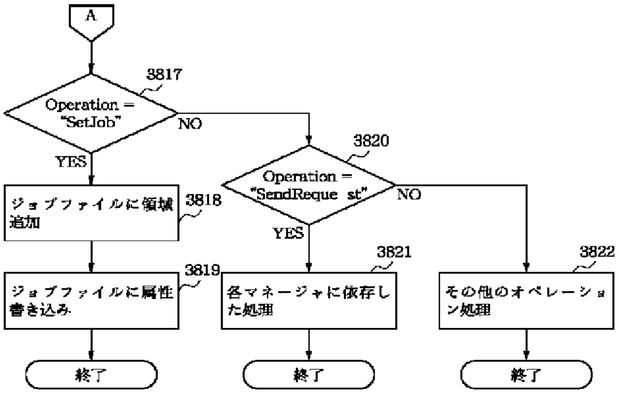
【図59】

フォントマネージャ，フォームオーバーレイマネージャ，ログマネージャ，カラープロファイルマネージャでのジョブ処理（アップロード）



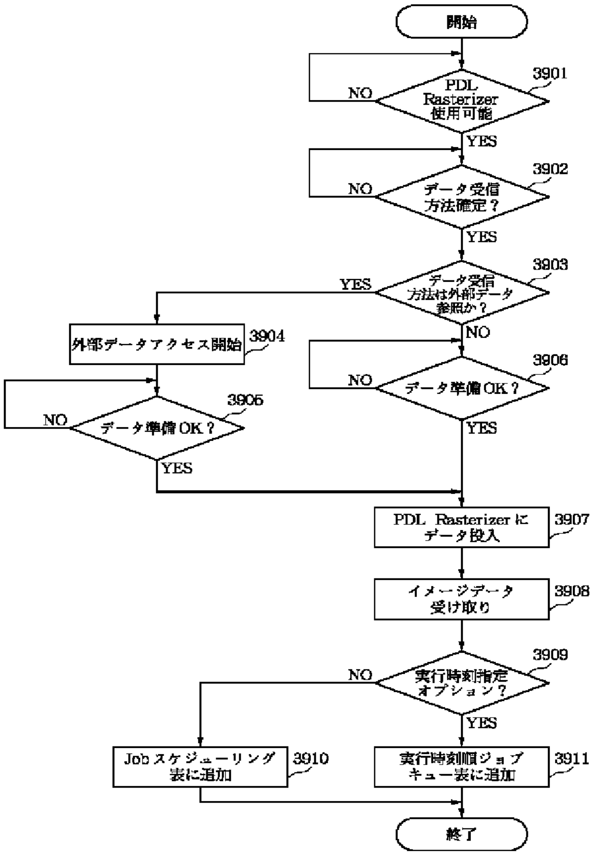
【図54】

各マネージャでのジョブスクリプト処理（続き）



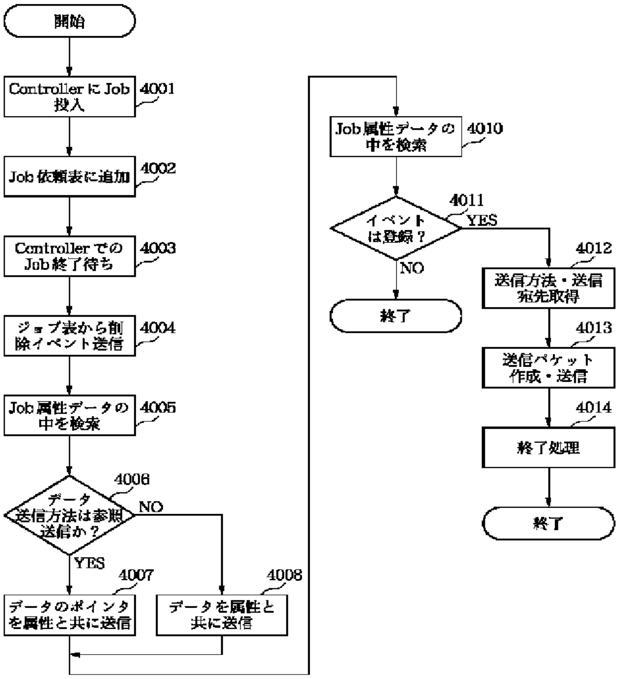
【図55】

プリントジョブマネージャでのジョブ処理
（データ受信からジョブスケジューリング表への登録まで）



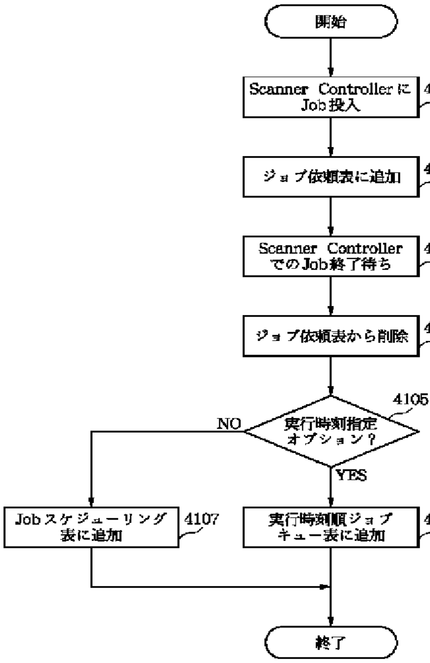
【図５６】

スキャンジョブマネージャでのジョブ処理



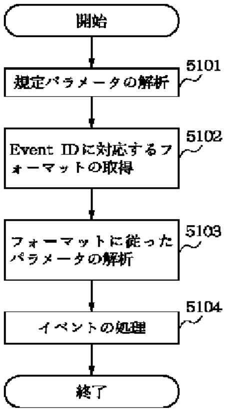
【図５７】

コピージョブマネージャでのジョブ処理


開始
Scanner Controller に Job 投入
4101
ジョブ依頼表に追加
4102
Scanner Controller での Job 終了待ち
4103
ジョブ依頼表から削除
4104
4105
実行時刻指定オプション？
NO
YES
Job スケジューリング表に追加
4107
実行時刻順ジョブキュー表に追加
4106
終了

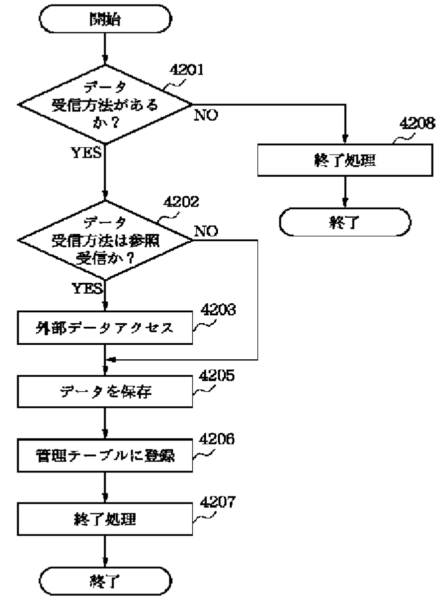
【図６９】

イベントの処理


開始
規定パラメータの解析
5101
Event ID に対応するフォーマットの取得
5102
フォーマットに従ったパラメータの解析
5103
イベントの処理
5104
終了

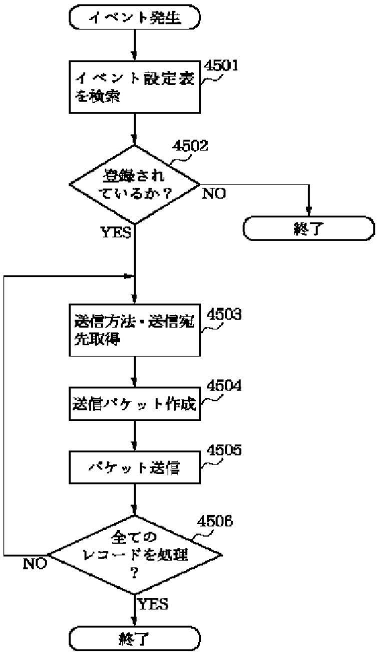
【図５８】

フォントマネージャ、フォームーオーバーレイマネージャ、ログマネージャ、カラープロファイルマネージャでのジョブ処理（ダウンロード）



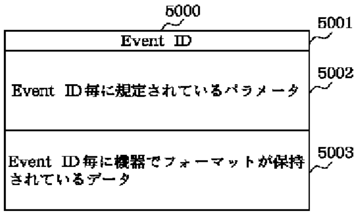
【図６１】

イベント送信



【図６８】

イベントの構造


5000
5001
Event ID
5002
Event ID 毎に規定されているパラメータ
5003
Event ID 毎に機器でフォーマットが保持されているデータ

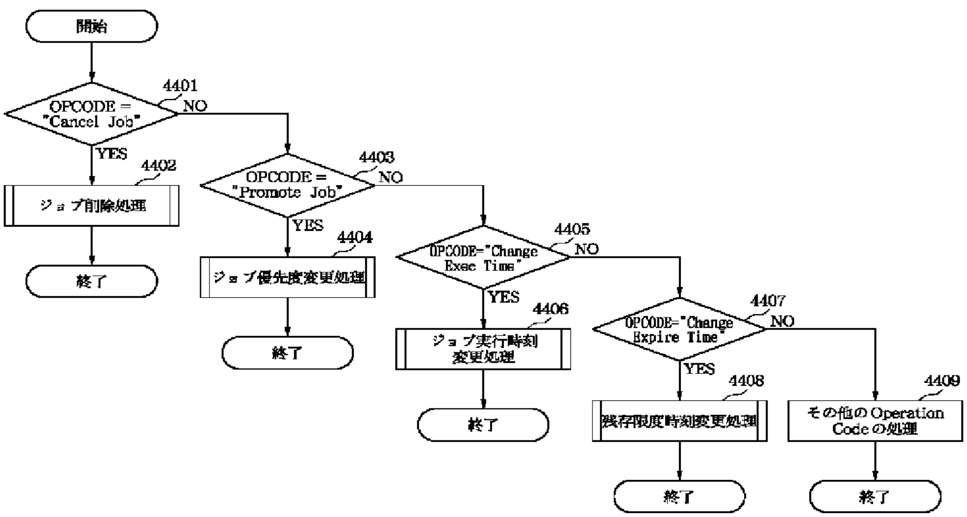
【図８４】

実行時刻順ジョブキュー表

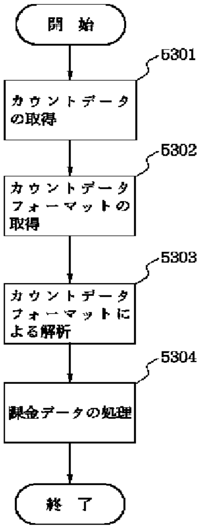
1561　　1562

指定実行時刻	JobID
13：45	13
14：00	12
14：30	15

【図６０】

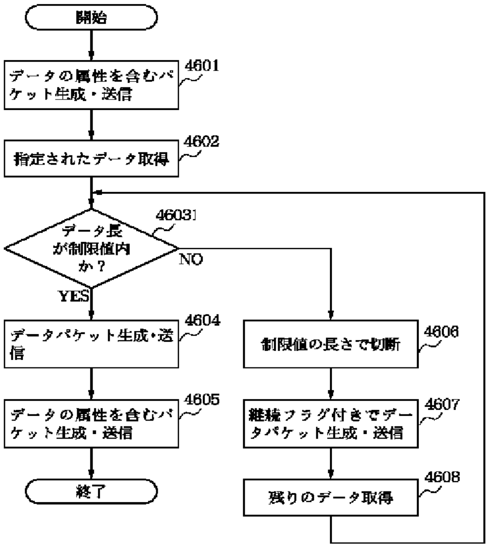

各マネージャでのジョブ管理

【図７１】

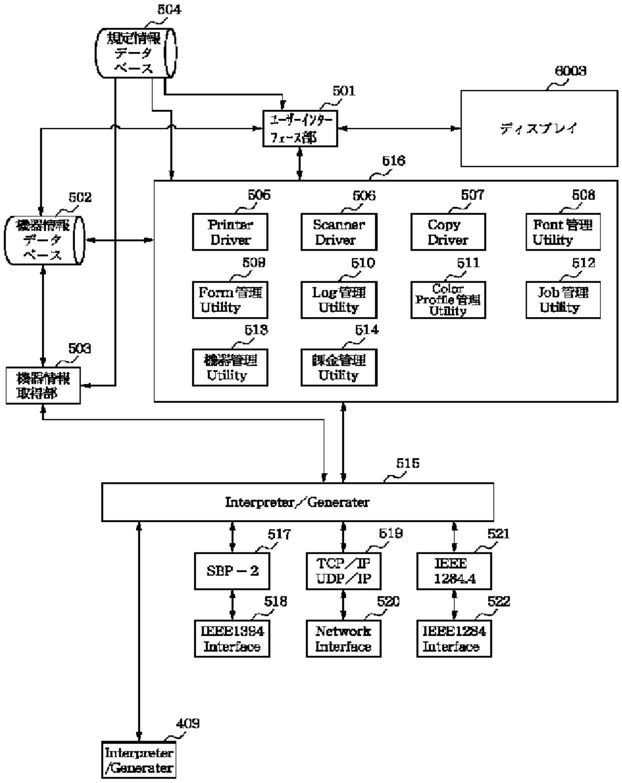

課金情報の取得・処理

開　始
5301
カウントデータの取得
5302
カウントデータフォーマットの取得
5303
カウントデータフォーマットによる解析
5304
課金データの処理
終　了

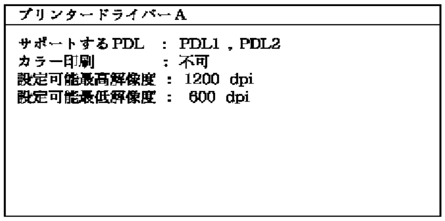
【図６２】


機器からのデータ（スクリプト）送信

【図６４】

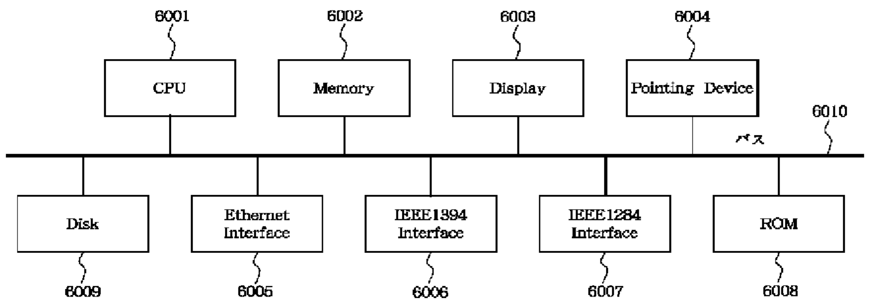


【図８１】


プリンタードライバーA

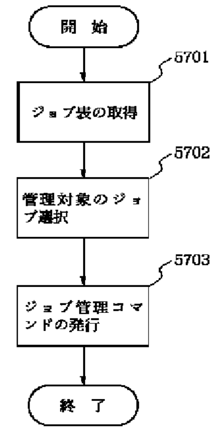
サポートするPDL　：　PDL1 , PDL2
カラー印刷　　　　：　不可
設定可能最高解像度　：　1200 dpi
設定可能最低解像度　：　600 dpi

【図６３】



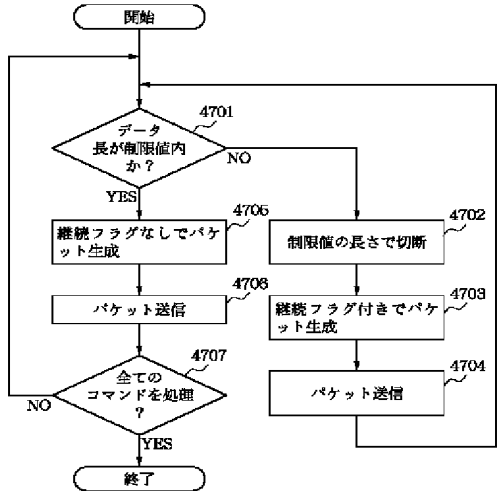
【図７５】

ジョブ管理コマンドの発行



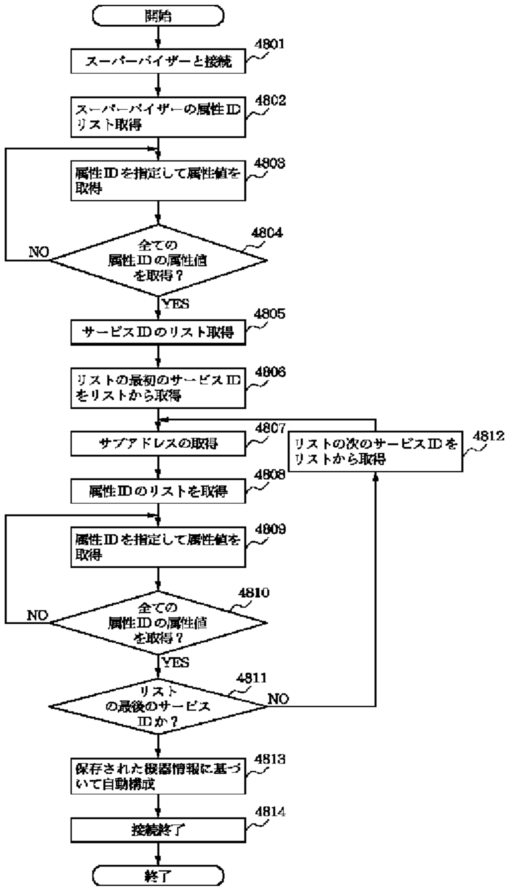
【図６５】

パケット生成・送信処理



【図６６】

クライアント側での情報取得およびソフトウエア自動構成


開始
4801
スーパーバイザーと接続
4802
スーパーバイザーの属性IDリスト取得
4803
属性IDを指定して属性値を取得
4804
全ての属性IDの属性値を取得？
NO
YES
4805
サービスIDのリスト取得
4806
リストの最初のサービスIDをリストから取得
4807
サブアドレスの取得
4812
リストの次のサービスIDをリストから取得
4808
属性IDのリストを取得
4809
属性IDを指定して属性値を取得
4810
全ての属性IDの属性値を取得？
NO
YES
4811
リストの最後のサービスIDか？
NO
4813
保存された機器情報に基づいて自動構成
4814
接続終了
終了

【図８２】

プリンタードライバーB

サポートするPDL　：PDL3
カラー印刷　　　　：可
設定可能最高解像度：600 dpi
設定可能最低解像度：300 dpi

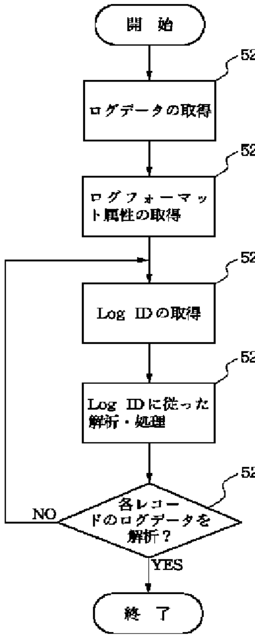
【図８３】

ジョブスケジューリング表

1551	優先度１のJobID	5	7	8		
1552	優先度２のJobID	2	3	4		
1553	優先度３のJobID	1	6			

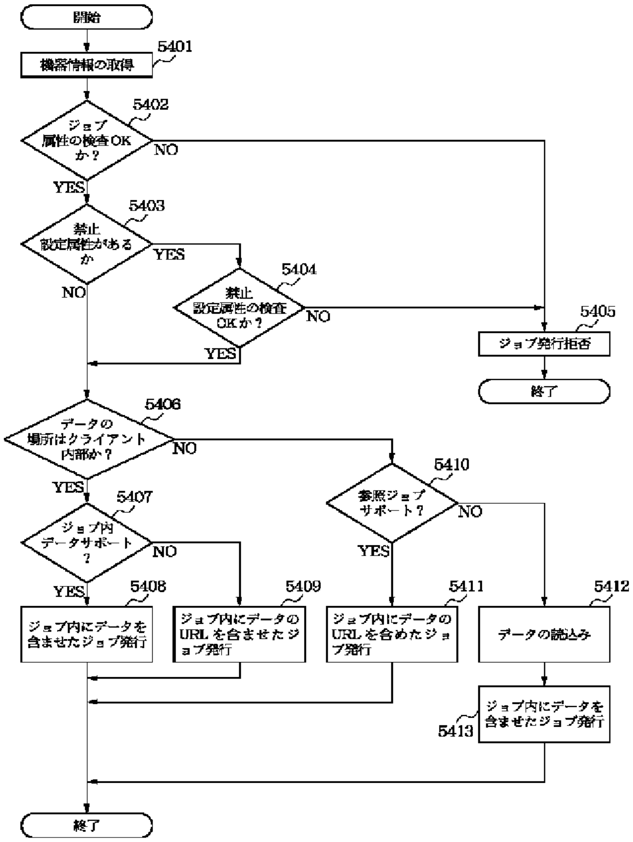
【図７０】

ログの処理



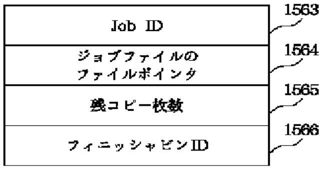
【図７２】

ジョブの発行（プリントジョブ、データダウンロード）



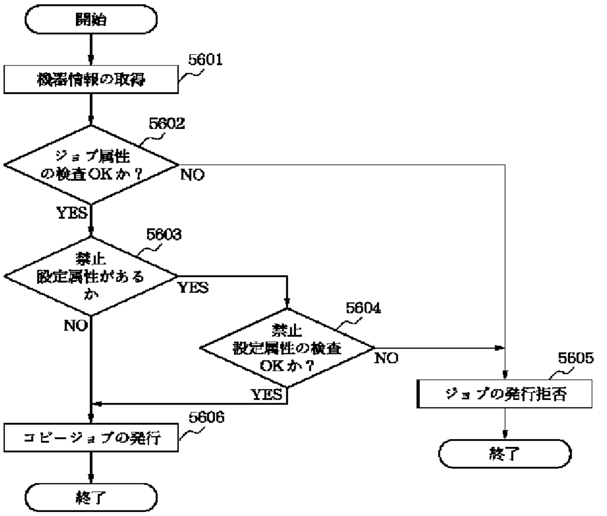
【図８５】

ジョブコンテキスト表

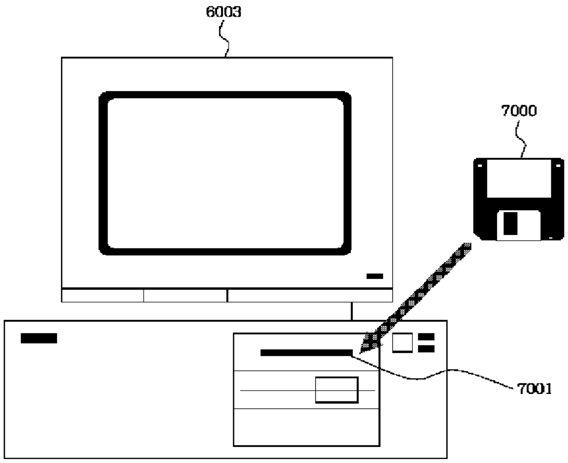


【図７４】

コピージョブの発行

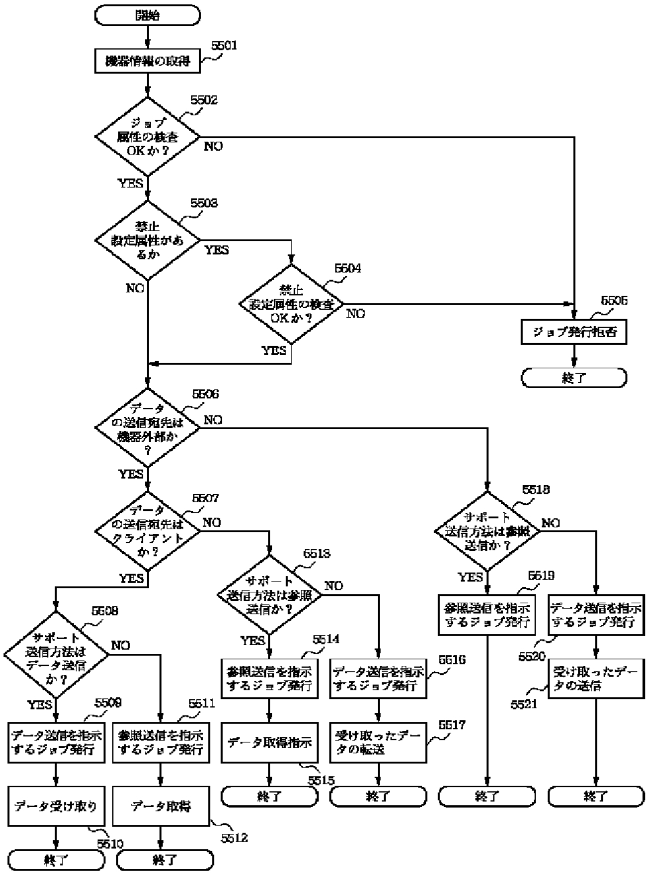


【図７８】



【図７３】

ジョブの発行（スキャンジョブ、データアップロード）

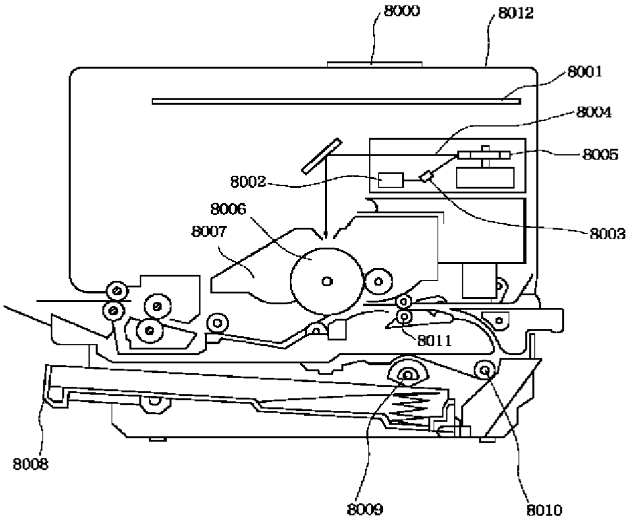


【図７６】

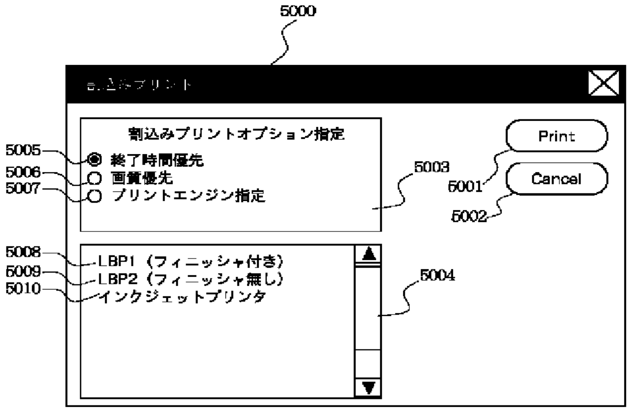
DISK315
UI Driver401
Network Interface Driver402
IEEE1284 Interface Driver403
IEEE1394 Interface Driver404
User Interface Manager405
TCP／IP,UDP／IP Module406
IEEE1284.4 Module407
SBB－2 Module408
Interpreter／Generater409
Supervisor410
Security Gate411
Dispatcher412
Font Manager413
Form Overlay Manager414
Log Manager415
Color Profile Manager416
PDL（PS）Rasterizer417
PDL（PCL）Rasterizer418
Scan Job Manager419
Scan Controler420
Print Job Manager501
Print Job Manager502
Print Job Manager503
Print Job Manager504
Print Job Manager505
Print Job Manager506
Print Job Manager507
Print Job Manager508
Print Job Manager509
LBP Controler510
LBP Controler511
Inkjet Controler512
Copy Job Manager601
Copy Job Manager602
Copy Job Manager603
Copy Job Manager604
Copy Job Manager605
Copy Job Manager606
Copy Job Manager607
Copy Job Manager608
図47に示す処理プログラム
図48に示す処理プログラム
図49に示す処理プログラム
図50に示す処理プログラム
図51に示す処理プログラム
図53に示す処理プログラム
図54に示す処理プログラム
図55に示す処理プログラム
図56に示す処理プログラム
図57に示す処理プログラム
図58に示す処理プログラム
図59に示す処理プログラム
図60に示す処理プログラム
図61に示す処理プログラム
図62に示す処理プログラム

図86、87、88に示す処理プログラム
図89に示す処理プログラム
図90に示す処理プログラム
図91に示す処理プログラム
図92に示す処理プログラム
図93、94、95に示す処理プログラム
その他のプログラムと各種データ

【図７９】



【図９６】



割込みプリントオプション指定

● 終了時間優先

○ 画質優先

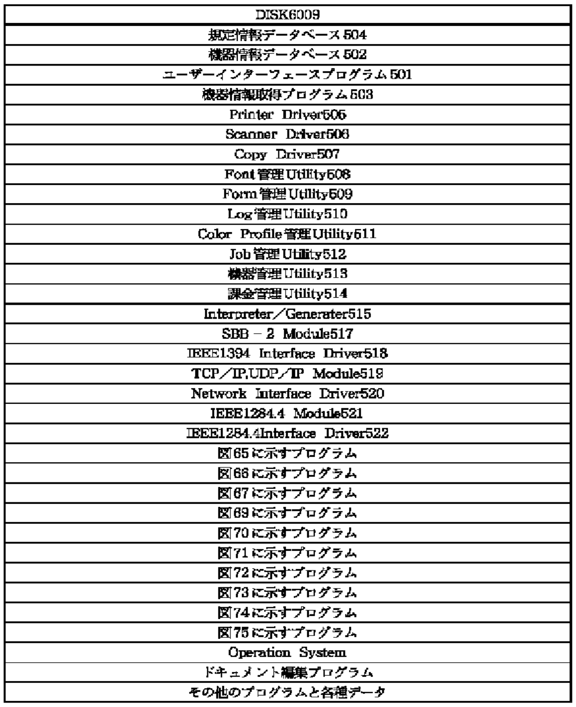
○ プリントエンジン指定

LBP1（フィニッシャ付き）

LBP2（フィニッシャ無し）

インクジェットプリンタ

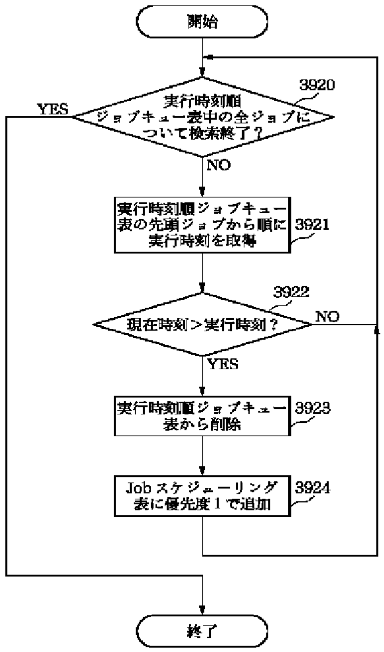
【図７７】



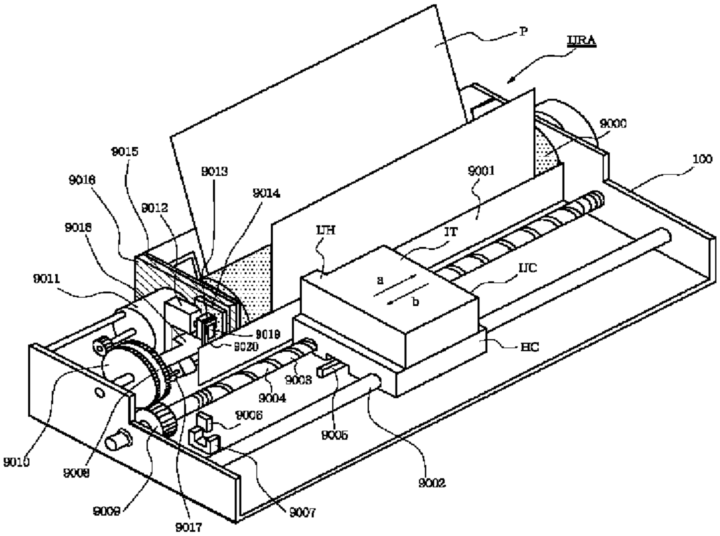
DISK6009
規定情報データベース 504
機器情報データベース 502
ユーザーインターフェースプログラム 501
機器情報取得プログラム 503
Printer Driver505
Scanner Driver506
Copy Driver507
Font 管理 Utility508
Form 管理 Utility509
Log 管理 Utility510
Color Profile 管理 Utility511
Job 管理 Utility512
機器管理 Utility513
課金管理 Utility514
Interpreter／Generater515
SBB－2 Module517
IEEE1394 Interface Driver518
TCP／IP,UDP／IP Module519
Network Interface Driver520
IEEE1284.4 Module521
IEEE1284.4Interface Driver522
図65 に示すプログラム
図66 に示すプログラム
図67 に示すプログラム
図69 に示すプログラム
図70 に示すプログラム
図71 に示すプログラム
図72 に示すプログラム
図73 に示すプログラム
図74 に示すプログラム
図75 に示すプログラム
Operation System
ドキュメント編集プログラム
その他のプログラムと各種データ

【図８６】

プリントジョブマネージャでのジョブ処理
（実行時刻順ジョブキュー表からジョブスケジューリング表への登録）

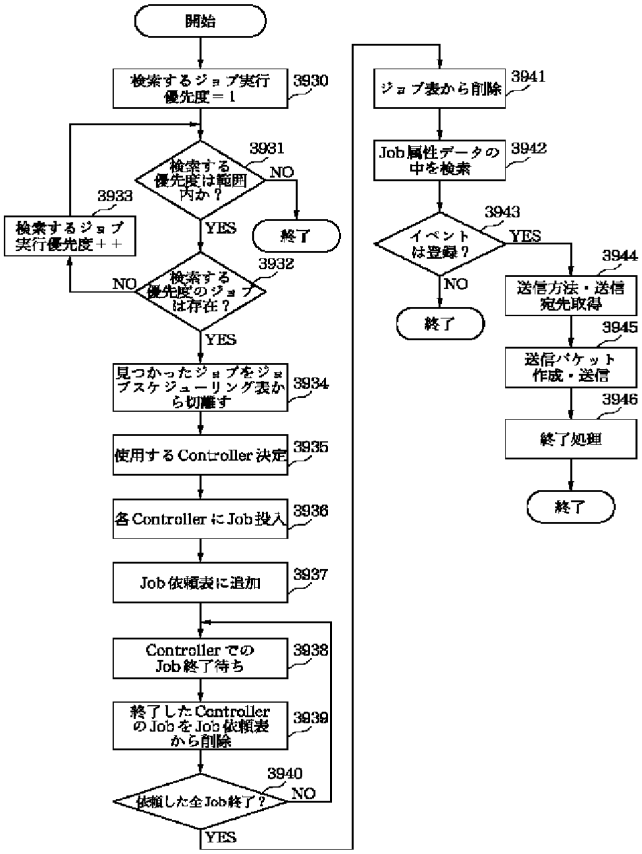

開始
実行時刻順ジョブキュー表中の全ジョブについて検索終了？
3920
YES
NO
実行時刻順ジョブキュー表の先頭ジョブから順に実行時刻を取得
3921
現在時刻＞実行時刻？
3922
NO
YES
実行時刻順ジョブキュー表から削除
3923
Job スケジューリング表に優先度１で追加
3924
終了

【図８０】

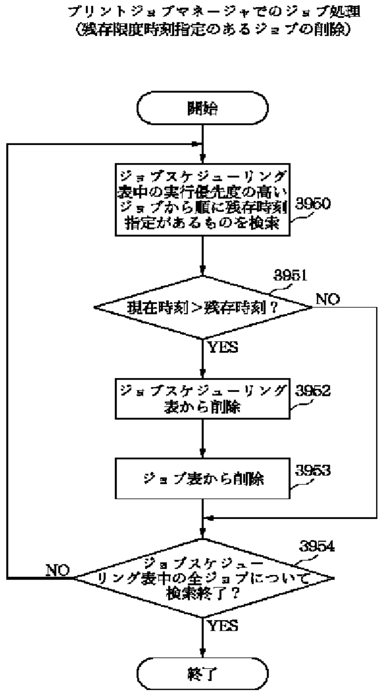


【図８７】

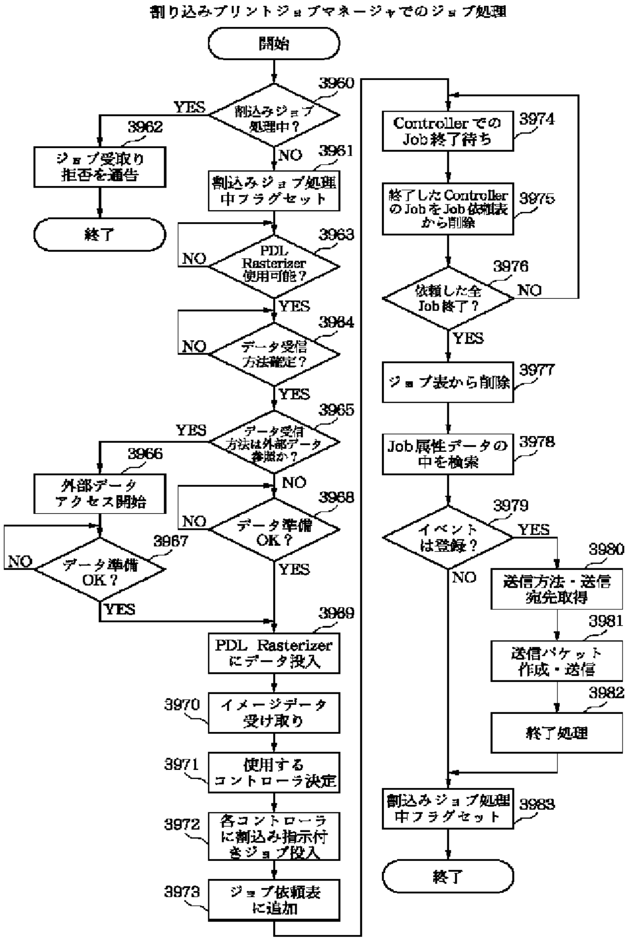
プリントジョブマネージャでのジョブ処理
（ジョブスケジューリング表から実行終了まで）



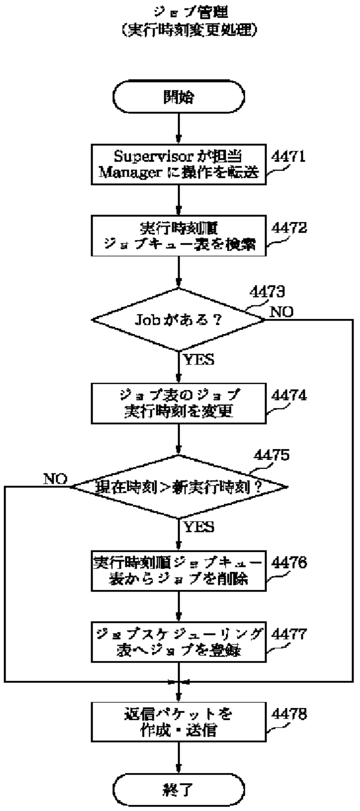
【図８８】


プリントジョブマネージャでのジョブ処理
（残存限度時刻指定のあるジョブの削除）
開始
ジョブスケジューリング表中の実行優先度の高いジョブから順に残存時刻指定があるものを検索
3950
現在時刻＞残存時刻？
3951
NO
YES
ジョブスケジューリング表から削除
3952
ジョブ表から削除
3953
ジョブスケジューリング表中の全ジョブについて検索終了？
3954
NO
YES
終了

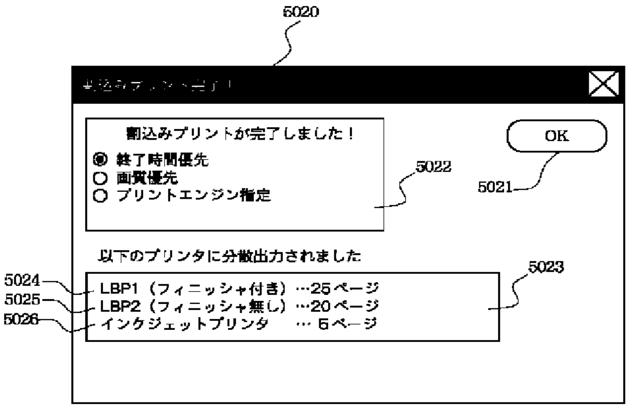
【図８９】



【図９４】


ジョブ管理
（実行時刻変更処理）
開始
Supervisor が担当 Manager に操作を転送
4471
実行時刻順ジョブキュー表を検索
4472
Job がある？
4473
NO
YES
ジョブ表のジョブ実行時刻を変更
4474
現在時刻＞新実行時刻？
4475
NO
YES
実行時刻順ジョブキュー表からジョブを削除
4476
ジョブスケジューリング表へジョブを登録
4477
返信パケットを作成・送信
4478
終了

【図９７】



割込みプリント完了！

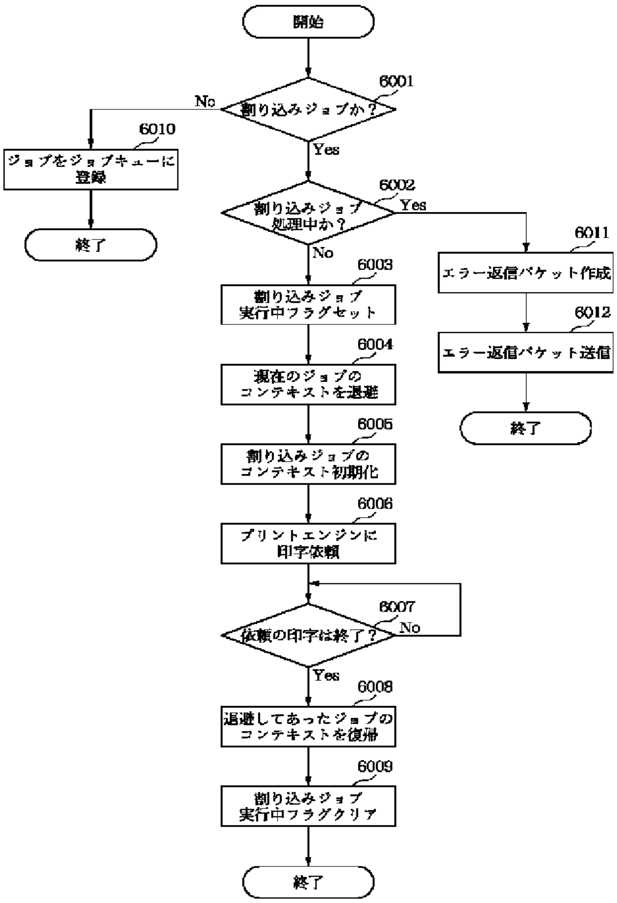
割込みプリントが完了しました！

● 終了時間優先
○ 画質優先
○ プリントエンジン指定

以下のプリンタに分散出力されました

LBP1（フィニッシャ付き）…25ページ
LBP2（フィニッシャ無し）…20ページ
インクジェットプリンタ　… 5ページ

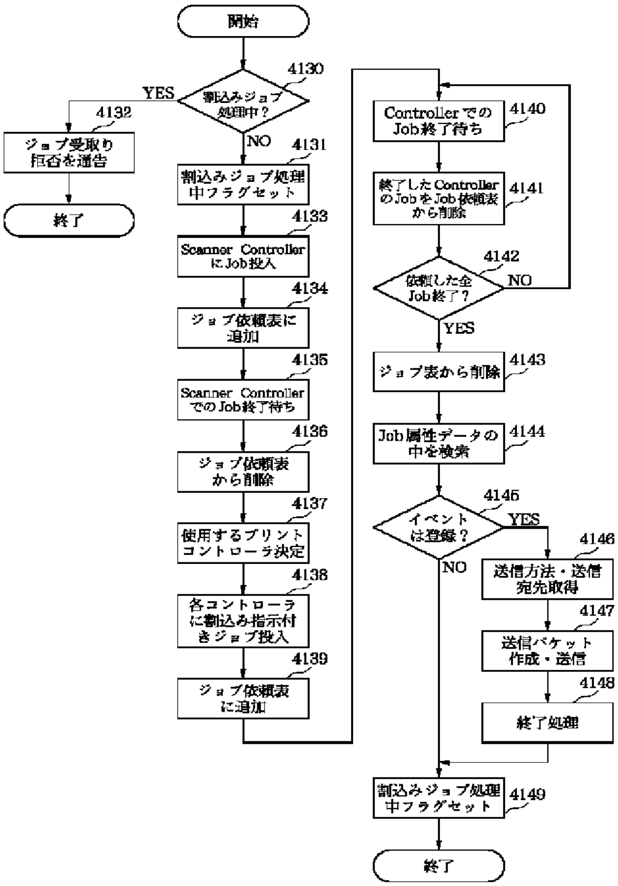
【図９０】



Print Controller のJob処理

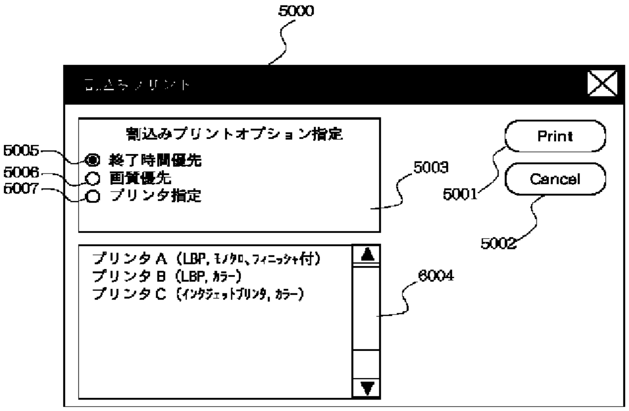
開始
6001
割り込みジョブか？
No
Yes
6010
ジョブをジョブキューに登録
終了
6002
割り込みジョブ処理中か？
Yes
No
6011
エラー返信パケット作成
6012
エラー返信パケット送信
終了
6003
割り込みジョブ実行中フラグセット
6004
現在のジョブのコンテキストを退避
6005
割り込みジョブのコンテキスト初期化
6006
プリントエンジンに印字依頼
6007
依頼の印字は終了？
No
Yes
6008
退避してあったジョブのコンテキストを復帰
6009
割り込みジョブ実行中フラグクリア
終了

【図９１】



割り込みコピージョブマネージャでのジョブ処理

【図９８】

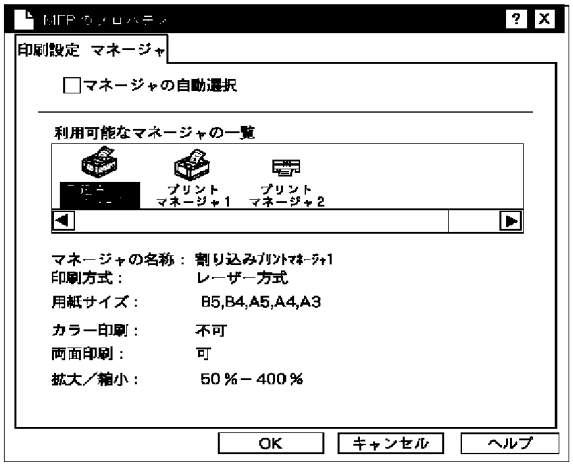


割込みプリントオプション指定

● 終了時間優先
○ 画質優先
○ プリンタ指定

プリンタＡ（LBP，ﾓﾉｸﾛ、ﾌｨﾆｯｼｬ付）
プリンタＢ（LBP，ｶﾗｰ）
プリンタＣ（ｲﾝｸｼﾞｪｯﾄﾌﾟﾘﾝﾀ，ｶﾗｰ）

【図９９】



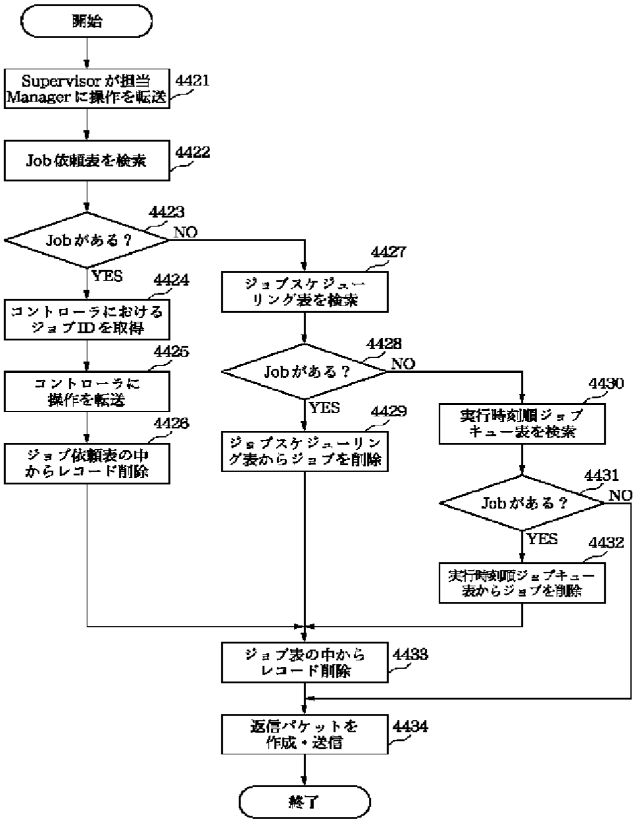
□マネージャの自動選択

利用可能なマネージャの一覧

マネージャの名称：　割り込みﾌﾟﾘﾝﾄﾏﾈｰｼﾞｬ1
印刷方式：　レーザー方式
用紙サイズ：　B5,B4,A5,A4,A3
カラー印刷：　不可
両面印刷：　可
拡大／縮小：　50％－400％

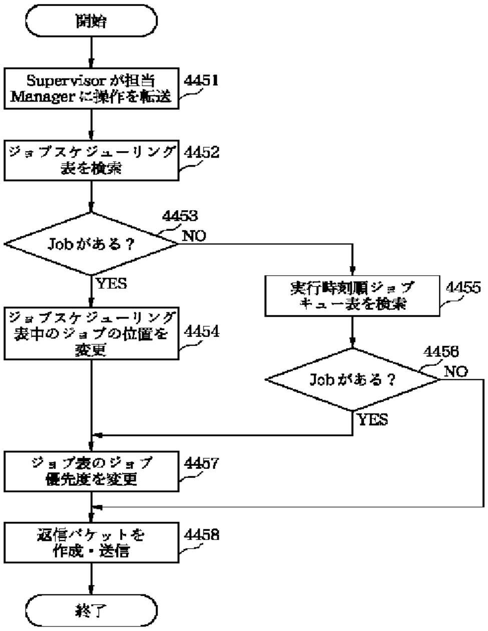
【図９２】

各マネージャでのジョブ管理
（ジョブ削除処理）

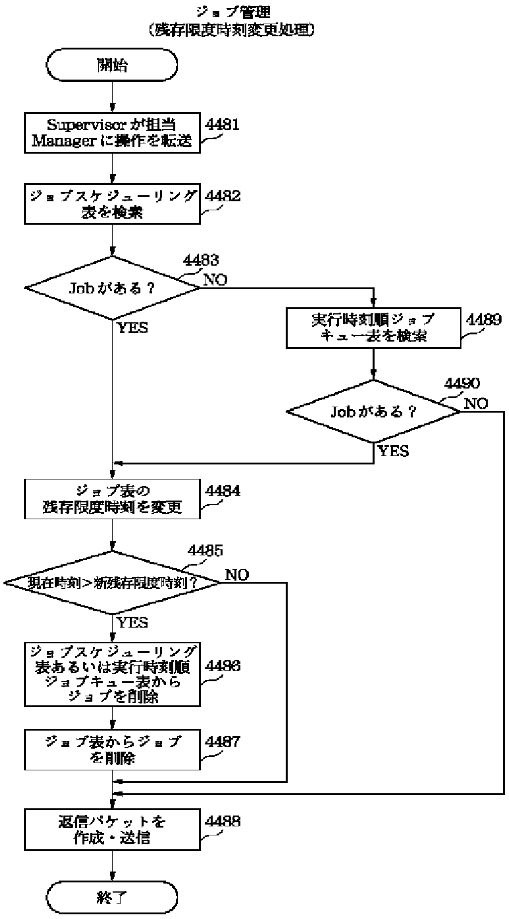

開始
Supervisor が担当 Manager に操作を転送
4421
Job 依頼表を検索
4422
Job がある？
4423
NO
YES
コントローラにおけるジョブ ID を取得
4424
コントローラに操作を転送
4425
ジョブ依頼表の中からレコード削除
4426
ジョブスケジューリング表を検索
4427
Job がある？
4428
NO
YES
ジョブスケジューリング表からジョブを削除
4429
実行時刻順ジョブキュー表を検索
4430
Job がある？
4431
NO
YES
実行時刻順ジョブキュー表からジョブを削除
4432
ジョブ表の中からレコード削除
4433
返信パケットを作成・送信
4434
終了

【図９３】

ジョブ管理
（ジョブ優先度変更処理）


開始
Supervisor が担当 Manager に操作を転送
4451
ジョブスケジューリング表を検索
4452
Job がある？
4453
NO
YES
ジョブスケジューリング表中のジョブの位置を変更
4454
実行時刻順ジョブキュー表を検索
4455
Job がある？
4456
NO
YES
ジョブ表のジョブ優先度を変更
4457
返信パケットを作成・送信
4458
終了

【図９５】


ジョブ管理
（残存限度時刻変更処理）
開始
Supervisorが担当Managerに操作を転送
4481
ジョブスケジューリング表を検索
4482
Jobがある？
4483
NO
YES
実行時刻順ジョブキュー表を検索
4489
Jobがある？
4490
NO
YES
ジョブ表の残存限度時刻を変更
4484
現在時刻＞新残存限度時刻？
4485
NO
YES
ジョブスケジューリング表あるいは実行時刻順ジョブキュー表からジョブを削除
4486
ジョブ表からジョブを削除
4487
返信パケットを作成・送信
4488
終了

フロントページの続き

(72)発明者　佐々木　安彦
東京都大田区下丸子３丁目30番２号キヤノン株式会社内

(72)発明者　長田　守
東京都大田区下丸子３丁目30番２号キヤノン株式会社内

(72)発明者　井上　敬史
東京都大田区下丸子３丁目30番２号キヤノン株式会社内

(72)発明者　下平　真子
東京都大田区下丸子３丁目30番２号キヤノン株式会社内

(72)発明者　高木　智子
東京都大田区下丸子３丁目30番２号キヤノン株式会社内

(54)【発明の名称】　周辺機器及び周辺機器制御方法及び周辺機器制御システム及び周辺機器制御プログラムを記憶した記憶媒体及び周辺機器制御プログラムを送出する送出装置及び周辺機器制御プログラム製品及び情報処理装置及び情報処理方法及び情報処理プログラムを記憶した記憶媒体及び情報処理プログラムを送出する送出装置及び情報処理プログラム製品

Disclaimer:
This English translation is produced by machine translation and may contain errors. The JPO, the INPIT, and those who drafted this document in the original language are not responsible for the result of the translation.

Notes:
1. Untranslatable words are replaced with asterisks (****).
2. Texts in the figures are not translated and shown as it is.

Translated: 11:03:42 JST 03/31/2010
Dictionary: Last updated 03/12/2010 / Priority:

FULL CONTENTS

[Claim(s)]

[Claim 1]Two or more device control programs which are the peripheral equipment connected with an information processor, hold a function of two or more device engines of said peripheral equipment, and manage a job in the device engine concerned, If an interruption job is inputted from said information processor while performing one of said two or more of the device control programs and processing a job, Peripheral equipment having a control means which chooses and executes other different device control programs from said device control program currently executed.

[Claim 2]The peripheral equipment according to claim 1 after evacuating an execution context of a job which was being performed when said interruption job was inputted and said device control program's completing execution of this interruption job, wherein it returns an execution context of said job which evacuated.

[Claim 3]The peripheral equipment according to claim 1 characterized by keeping a control means from making other interruption jobs input into multiplex during interruption job execution.

[Claim 4]The peripheral equipment according to claim 1, wherein said interruption job is an interruption print job.

[Claim 5]The peripheral equipment according to claim 1, wherein said interruption job is an interruption copy job.

[Claim 6]When said interruption copy job is inputted, [said control means] After a job is supplied to a scanner control program which holds a function of a scanner engine of said peripheral equipment, and manages a job in the scanning engine concerned, The peripheral equipment according to claim 5 choosing arbitrary printer control programs which hold a function of printer engine of said peripheral equipment, and manage a job in the printer engine concerned.

[Claim 7]The peripheral equipment according to claim 1, wherein said device control program is a scanner control program which controls a scanner engine of said peripheral equipment.

[Claim 8]The peripheral equipment according to claim 1, wherein said device control program is a laser beam printer control program which controls a laser beam printer engine of said peripheral equipment.

[Claim 9]The peripheral equipment according to claim 1, wherein said device control program is an ink-jet printer control program which controls an ink-jet printer engine of said peripheral equipment.
[Claim 10]The peripheral equipment according to claim 1, wherein said peripheral equipment has a print job control program and a copy job control program which control said device control program, and a scanning job control program.
[Claim 11]It is the peripheral equipment control method in peripheral equipment connected with an information processor, While performing one of two or more of the device control programs which hold a function of two or more device engines of said peripheral equipment, and manage a job in the device engine concerned and processing a job, A peripheral equipment control method choosing and executing other different device control programs from said device control program currently executed if an interruption job is inputted from said information processor.
[Claim 12]A peripheral equipment control method according to claim 11 after evacuating an execution context of a job which was being performed when said interruption job was inputted and said device control program's completing execution of this interruption job, wherein it returns an execution context of said job which evacuated.
[Claim 13]A peripheral equipment control method according to claim 11 characterized by making it not make other interruption jobs input into multiplex during said interruption job execution.
[Claim 14]A peripheral equipment control method according to claim 11, wherein said interruption job is an interruption print job.
[Claim 15]A peripheral equipment control method according to claim 11, wherein said interruption job is an interruption copy job.
[Claim 16]When said interruption copy job is inputted, After a job is supplied to a scanner control program which holds a function of a scanner engine of said peripheral equipment, and manages a job in the scanning engine concerned, A peripheral equipment control method according to claim 15 choosing arbitrary printer control programs which hold a function of printer engine of said peripheral equipment, and manage a job in the printer engine concerned.
[Claim 17]A peripheral equipment control method according to claim 11, wherein said device control program is a scanner control program which controls a scanner engine of said peripheral equipment.
[Claim 18]A peripheral equipment control method according to claim 11, wherein said device control program is a laser beam printer control program which controls a laser beam printer engine of said peripheral equipment.
[Claim 19]A peripheral equipment control method according to claim 11, wherein said device control program is an ink-jet printer control program which controls an ink-jet printer engine of said peripheral equipment.
[Claim 20]A peripheral equipment control method according to claim 11, wherein said peripheral equipment has a print job control program and a copy job control program which control said device control program, and a scanning job control program.
[Claim 21]It is the storage which memorized a peripheral equipment control program executed with peripheral equipment connected with an information processor, While performing one of two or more

of the device control programs which hold a function of two or more device engines of said peripheral equipment, and manage a job in the device engine concerned and processing a job, A storage memorizing a peripheral equipment control program which chooses and executes other different device control programs from said device control program currently executed if an interruption job is inputted from said information processor.
[Claim 22]The storage according to claim 21 after evacuating an execution context of a job which was being performed when said interruption job was inputted and said device control program's completing execution of this interruption job, wherein it returns an execution context of said job which evacuated.
[Claim 23]The storage according to claim 21 characterized by making it not make other interruption jobs input into multiplex during said interruption job execution.
[Claim 24]The storage according to claim 21, wherein said interruption job is an interruption print job.
[Claim 25]The storage according to claim 21, wherein said interruption job is an interruption copy job.
[Claim 26]When said interruption copy job is inputted, After a job is supplied to a scanner control program which holds a function of a scanner engine of said peripheral equipment, and manages a job in the scanning engine concerned, The storage according to claim 25 choosing arbitrary printer control programs which hold a function of printer engine of said peripheral equipment, and manage a job in the printer engine concerned.
[Claim 27]The storage according to claim 21, wherein said device control program is a scanner control program which controls a scanner engine of said peripheral equipment.
[Claim 28]The storage according to claim 21, wherein said device control program is a laser beam printer control program which controls a laser beam printer engine of said peripheral equipment.
[Claim 29]The storage according to claim 21, wherein said device control program is an ink-jet printer control program which controls an ink-jet printer engine of said peripheral equipment.
[Claim 30]The storage according to claim 21, wherein said peripheral equipment has a print job control program and a copy job control program which control said device control program, and a scanning job control program.
[Claim 31]An output means which is a peripheral equipment control system which comprises an information processor and peripheral equipment, and outputs a job to said peripheral equipment, Two or more device control programs which hold a function of two or more device engines of said peripheral equipment, and manage a job in the device engine concerned, If it interrupts from said output means and a job is outputted to said information processor while performing one of said two or more of the device control programs and processing a job, A peripheral equipment control system having a control means which chooses and executes other different device control programs from said device control program currently executed.
[Claim 32]The peripheral equipment control system according to claim 31 after evacuating an execution context of a job which was being performed when said interruption job was inputted and said device control program's completing execution of this interruption job, wherein it returns an execution context of said job which evacuated.

[Claim 33]The peripheral equipment control system according to claim 31 characterized by keeping a control means from making other interruption jobs input into multiplex during interruption job execution.
[Claim 34]The peripheral equipment control system according to claim 31, wherein said interruption job is an interruption print job.
[Claim 35]The peripheral equipment control system according to claim 31, wherein said interruption job is an interruption copy job.
[Claim 36]When said interruption copy job is inputted, [said control means] After a job is supplied to a scanner control program which holds a function of a scanner engine of said peripheral equipment, and manages a job in the scanning engine concerned, The peripheral equipment control system according to claim 35 choosing arbitrary printer control programs which hold a function of printer engine of said peripheral equipment, and manage a job in the printer engine concerned.
[Claim 37]The peripheral equipment control system according to claim 31, wherein said device control program is a scanner control program which controls a scanner engine of said peripheral equipment.
[Claim 38]The peripheral equipment control system according to claim 31, wherein said device control program is a laser beam printer control program which controls a laser beam printer engine of said peripheral equipment.
[Claim 39]The peripheral equipment control system according to claim 31, wherein said device control program is an ink-jet printer control program which controls an ink-jet printer engine of said peripheral equipment.
[Claim 40]The peripheral equipment control system according to claim 31, wherein said peripheral equipment has a print job control program and a copy job control program which control said device control program, and a scanning job control program.
[Claim 41]It is a sending device which sends out a peripheral equipment control program executed with peripheral equipment connected with an information processor, While performing one of two or more of the device control programs which hold a function of two or more device engines of said peripheral equipment, and manage a job in the device engine concerned and processing a job, A sending device sending out a peripheral equipment control program which chooses and executes other different device control programs from said device control program currently executed if an interruption job is inputted from said information processor.
[Claim 42]The sending device according to claim 41 after evacuating an execution context of a job which was being performed when said interruption job was inputted and said device control program's completing execution of this interruption job, wherein it returns an execution context of said job which evacuated.
[Claim 43]The sending device according to claim 41 characterized by making it not make other interruption jobs input into multiplex during said interruption job execution.
[Claim 44]The sending device according to claim 41, wherein said interruption job is an interruption print job.
[Claim 45]The sending device according to claim 41, wherein said interruption job is an interruption

copy job.

[Claim 46]When said interruption copy job is inputted, After a job is supplied to a scanner control program which holds a function of a scanner engine of said peripheral equipment, and manages a job in the scanning engine concerned, The sending device according to claim 45 choosing arbitrary printer control programs which hold a function of printer engine of said peripheral equipment, and manage a job in the printer engine concerned.

[Claim 47]The sending device according to claim 41, wherein said device control program is a scanner control program which controls a scanner engine of said peripheral equipment.

[Claim 48]The sending device according to claim 41, wherein said device control program is a laser beam printer control program which controls a laser beam printer engine of said peripheral equipment.

[Claim 49]The sending device according to claim 41, wherein said device control program is an ink-jet printer control program which controls an ink-jet printer engine of said peripheral equipment.

[Claim 50]The sending device according to claim 41, wherein said peripheral equipment has a print job control program and a copy job control program which control said device control program, and a scanning job control program.

[Claim 51]They are the peripheral equipment control program products performed with peripheral equipment connected with an information processor, While performing one of two or more of the device control programs which hold a function of two or more device engines of said peripheral equipment, and manage a job in the device engine concerned and processing a job, Peripheral equipment control program products choosing and executing other different device control programs from said device control program currently executed if an interruption job is inputted from said information processor.

[Claim 52]After said device control program evacuates an execution context of a job which was being performed when said interruption job was inputted and execution of this interruption job is completed, The peripheral equipment control program products according to claim 51 returning an execution context of said job which evacuated.

[Claim 53]The peripheral equipment control program product according to claim 51 characterized by making it not make other interruption jobs input into multiplex during said interruption job execution.

[Claim 54]The peripheral equipment control program product according to claim 51, wherein said interruption job is an interruption print job.

[Claim 55]The peripheral equipment control program product according to claim 51, wherein said interruption job is an interruption copy job.

[Claim 56]When said interruption copy job is inputted, After a job is supplied to a scanner control program which holds a function of a scanner engine of said peripheral equipment, and manages a job in the scanning engine concerned, The peripheral equipment control program products according to claim 55 choosing arbitrary printer control programs which hold a function of printer engine of said peripheral equipment, and manage a job in the printer engine concerned.

[Claim 57]The peripheral equipment control program product according to claim 51, wherein said device control program is a scanner control program which controls a scanner engine of said peripheral equipment.

[Claim 58]The peripheral equipment control program product according to claim 51, wherein said device control program is a laser beam printer control program which controls a laser beam printer engine of said peripheral equipment.

[Claim 59]The peripheral equipment control program product according to claim 51, wherein said device control program is an ink-jet printer control program which controls an ink-jet printer engine of said peripheral equipment.

[Claim 60]The peripheral equipment control program product according to claim 51, wherein said peripheral equipment has a print job control program and a copy job control program which control said device control program, and a scanning job control program.

[Claim 61]The peripheral equipment control system according to claim 31 having a displaying means which displays discernment information for identifying said device control program.

[Claim 62]The peripheral equipment control system according to claim 61, wherein said displaying means expresses said device control program by an icon.

[Claim 63]The peripheral equipment control system according to claim 62, wherein said displaying means displays an icon expressing said device control program chosen and executed when an interruption job is outputted.

[Claim 64]A peripheral equipment control system of Claim 61, wherein said displaying means displays information on a device engine which said device control program chosen and executed when an interruption job is outputted manages.

[Claim 65]A peripheral equipment system of Claim 61, wherein said displaying means displays a function which said device control program chosen and executed when an interruption job is outputted has.

[Claim 66]Said displaying means displays an icon expressing said two or more device control programs in a list, and. [a displaying means] The peripheral equipment control system according to claim 62 characterized by what is displayed so that an icon of said device control program chosen and executed may be discriminated from an icon of said other device control programs, when an interruption job is outputted.

[Claim 67]The peripheral equipment control system according to claim 61 with which information which identifies said device control program in which said displaying means is chosen and performed when an interruption job is directed and an interruption job is outputted, and it deals is characterized by plurality and displayed in a list.

[Claim 68]The peripheral equipment control system according to claim 61, wherein it has a selecting means for choosing either of the discernment information currently shown by said displaying means and an account of a last output means outputs a job indicating a device control program corresponding to discernment information with said selected selecting means.

[Claim 69]An information processor which outputs a job to peripheral equipment, comprising:

An acquisition means which acquires information about a device control program which holds a

function of a device engine in said peripheral equipment, and manages a job in the device engine concerned.
A displaying means which displays discernment information which identifies said device control program based on information acquired by said acquisition means.
A selecting means for choosing discernment information displayed by said displaying means.
An output means which outputs a job indicating a device control program corresponding to discernment information with said selected selecting means.

[Claim 70]The information processor according to claim 69, wherein said displaying means displays discernment information which identifies a device control program chosen and executed when an interruption job is outputted.
[Claim 71]An information processor of Claim 69, wherein said displaying means displays information on a device engine which said device control program chosen and executed when an interruption job is outputted manages.
[Claim 72]An information processor of Claim 69, wherein said displaying means displays a function which said device control program chosen and executed when an interruption job is outputted has.
[Claim 73]Said displaying means displays an icon expressing said two or more device control programs in a list, and. [a displaying means] The information processor according to claim 69 characterized by what is displayed so that an icon of said device control program chosen and executed may be discriminated from an icon of said other device control programs, when an interruption job is outputted.
[Claim 74]An information processing method which outputs a job to peripheral equipment, comprising:
An acquisition step which acquires information about a device control program which holds a function of a device engine in said peripheral equipment, and manages a job in the device engine concerned.
A display step which displays discernment information which identifies said device control program on a displaying means based on information acquired by said acquisition means.
A selection step for choosing discernment information displayed by said displaying means.
An output step which outputs a job indicating a device control program corresponding to discernment information with said selected selecting means.

[Claim 75]The information processing method according to claim 74 displaying discernment information which identifies a device control program chosen and executed in said display step when an interruption job is outputted.
[Claim 76]An information processing method of Claim 74 displaying information on a device engine which said device control program chosen and executed in said display step when an interruption job is outputted manages.
[Claim 77]An information processing method of Claim 74 displaying a function which said device control program chosen and executed in said display step when an interruption job is outputted has.
[Claim 78]In said display step, display an icon expressing said two or more device control programs in

a list, and. The information processing method according to claim 74 characterized by what is displayed so that an icon of said device control program chosen and executed may be discriminated from an icon of said other device control programs, when an interruption job is outputted.
[Claim 79]A recording medium characterized by comprising the following which stored an information processing program which outputs a job to peripheral equipment and in which reading in a computer is possible.
An acquisition step which acquires information about a device control program which the information processing program concerned holds a function of a device engine in said peripheral equipment, and manages a job in the device engine concerned.
A display step which displays discernment information which identifies said device control program on a displaying means based on information acquired by said acquisition means.
A selection step for choosing discernment information displayed by said displaying means.
An output step which outputs a job indicating a device control program corresponding to discernment information with said selected selecting means.

[Claim 80]The recording medium according to claim 79 displaying discernment information which identifies a device control program chosen and executed in said display step when an interruption job is outputted.
[Claim 81]A recording medium of Claim 79 displaying information on a device engine which said device control program chosen and executed in said display step when an interruption job is outputted manages.
[Claim 82]A recording medium of Claim 79 displaying a function which said device control program chosen and executed in said display step when an interruption job is outputted has.
[Claim 83]In said display step, display an icon expressing said two or more device control programs in a list, and. The recording medium according to claim 79 characterized by what is displayed so that an icon of said device control program chosen and executed may be discriminated from an icon of said other device control programs, when an interruption job is outputted.
[Claim 84]Are an information processing program which outputs a job to peripheral equipment a sending device to send out, and, [the information processing program concerned] An acquisition step which acquires information about a device control program which holds a function of a device engine in said peripheral equipment, and manages a job in the device engine concerned, A display step which displays discernment information which identifies said device control program on a displaying means based on information acquired by said acquisition means, A sending device having a selection step for choosing discernment information displayed by said displaying means, and an output step which outputs a job indicating a device control program corresponding to discernment information with said selected selecting means.
[Claim 85]The sending device according to claim 84 displaying discernment information which identifies a device control program chosen and executed in said display step when an interruption job is outputted.

[Claim 86]A sending device of Claim 84 displaying information on a device engine which said device control program chosen and executed in said display step when an interruption job is outputted manages.
[Claim 87]A sending device of Claim 84 displaying a function which said device control program chosen and executed in said display step when an interruption job is outputted has.
[Claim 88]In said display step, display an icon expressing said two or more device control programs in a list, and. The sending device according to claim 84 characterized by what is displayed so that an icon of said device control program chosen and executed may be discriminated from an icon of said other device control programs, when an interruption job is outputted.
[Claim 89]Information processing program products which output a job to peripheral equipment, comprising:
An acquisition step which acquires information about a device control program which the information processing program concerned holds a function of a device engine in said peripheral equipment, and manages a job in the device engine concerned.
A display step which displays discernment information which identifies said device control program on a displaying means based on information acquired by said acquisition means.
A selection step for choosing discernment information displayed by said displaying means.
An output step which outputs a job indicating a device control program corresponding to discernment information with said selected selecting means.

[Claim 90]The information processing program products according to claim 89 displaying discernment information which identifies a device control program chosen and executed in said display step when an interruption job is outputted.
[Claim 91]Information processing program products of Claim 89 displaying information on a device engine which said device control program chosen and executed in said display step when an interruption job is outputted manages.
[Claim 92]Information processing program products of Claim 89 displaying a function which said device control program chosen and executed in said display step when an interruption job is outputted has.
[Claim 93]In said display step, display an icon expressing said two or more device control programs in a list, and. The information processing program product according to claim 89 characterized by what is displayed so that an icon of said device control program chosen and executed may be discriminated from an icon of said other device control programs, when an interruption job is outputted.

[Detailed Description of the Invention]
[0001]

[Field of the Invention]This invention relates to the storage which memorized the peripheral equipment, the peripheral equipment control method, peripheral equipment control system, and peripheral equipment control programs which are controlled from an information processor via the connection form of a network and IEEE 1394 grade, such as a printer, a scanner, fax, and a copy machine.

[0002]

[Description of the Prior Art]When a job was supplied from a host computer (information processor) to the printer, the scanner, the fax, and the copy machine which were connected to the network etc., by the apparatus which carries mass memory storage, it became possible to accumulate many jobs in an inside simultaneously. Usually, a job is processed by the first in first serve (FIFS) method based on the order of the injection of the job.

[0003]

[Problem to be solved by the invention]An interruption job was not able to be controlled in conventional peripheral equipment and the multifunctional peripheral equipment which has a function of a printer, a scanner, fax, and a copy machine in one case especially.

[0004]The purpose of this invention is to provide the storage which memorized the user-friendly various functions peripheral equipment and the peripheral equipment control method for a user, the peripheral equipment control system, and the peripheral equipment control program.

[0005]

[Means for solving problem][peripheral equipment of this invention for attaining the above-mentioned purpose] Two or more device control programs which are the peripheral equipment connected with an information processor, hold a function of two or more device engines of said peripheral equipment, and manage a job in the device engine concerned, While performing one of said two or more of the device control programs and processing a job, when an interruption job is inputted from said information processor, it has a control means which chooses and executes other different device control programs from said device control program currently executed.

[0006][a peripheral equipment control method of this invention for attaining the above-mentioned purpose] It is the peripheral equipment control method in peripheral equipment connected with an information processor, While performing one of two or more of the device control programs which hold a function of two or more device engines of said peripheral equipment, and manage a job in the device engine concerned and processing a job, If an interruption job is inputted from said information processor, other different device control programs from said device control program currently executed will be chosen and executed.

[0007][a storage of this invention for attaining the above-mentioned purpose] It is the storage which memorized a peripheral equipment control program executed with peripheral equipment connected with an information processor, While performing one of two or more of the device control programs which hold a function of two or more device engines of said peripheral equipment, and manage a job in the device engine concerned and processing a job, If an interruption job is inputted from said information processor, a peripheral equipment control program which chooses and executes other

different device control programs from said device control program currently executed will be memorized.
[0008][a peripheral equipment control system of this invention for attaining the above-mentioned purpose] An output means which is a peripheral equipment control system which comprises an information processor and peripheral equipment, and outputs a job to said peripheral equipment, Two or more device control programs which hold a function of two or more device engines of said peripheral equipment, and manage a job in the device engine concerned, If it interrupts from said output means and a job is outputted to said information processor while performing one of said two or more of the device control programs and processing a job, It has a control means which chooses and executes other different device control programs from said device control program currently executed.

[0009][a sending device of this invention for attaining the above-mentioned purpose] It is a sending device which sends out a peripheral equipment control program executed with peripheral equipment connected with an information processor, While performing one of two or more of the device control programs which hold a function of two or more device engines of said peripheral equipment, and manage a job in the device engine concerned and processing a job, If an interruption job is inputted from said information processor, a peripheral equipment control program which chooses and executes other different device control programs from said device control program currently executed is sent out.
[0010][peripheral equipment control program products of this invention for attaining the above-mentioned purpose] They are the peripheral equipment control program products performed with peripheral equipment connected with an information processor, While performing one of two or more of the device control programs which hold a function of two or more device engines of said peripheral equipment, and manage a job in the device engine concerned and processing a job, If an interruption job is inputted from said information processor, other different device control programs from said device control program currently executed will be chosen and executed.
[0011][an information processor of this invention for entertaining the above-mentioned purpose] Are an information processor which outputs a job to peripheral equipment, and information about a device control program which holds a function of a device engine in said peripheral equipment, and manages a job in the device engine concerned is acquired, It is based on acquired information and a job which displayed discernment information which identifies a device control program chosen and executed, and pointed to a device control program corresponding to selected discernment information when an interruption job was outputted especially, said device control program and is outputted.
[0012]
[Mode for carrying out the invention](Composition of multifunctional peripheral equipment) Drawing 1 is a figure showing composition of multifunctional peripheral equipment (MFT:Multi-Function Peripheral) in which one embodiment of this invention is shown.
[0013]In drawing 1, 101 is a controller for controlling multifunctional peripheral equipment, has hardware constitutions shown in drawing 3, and has software (control program) composition as

shown in DISK315 as a storage of drawing 76. 102 is a scanner engine and is controlled by the controller 101. 103 and 104 are the laser beam printer engines (LBP) 1 and 2, and are controlled by the controller 101. The laser beam printer engine 1 of 103 is connected to the finisher 106, and it is possible to carry out staple (stapler) processing of two or more recording media (for example, paper) outputted from the printer engine collectively. The finisher 106 is also controlled by the controller 101. Also as for color printing, 105 is possible with an ink-jet-print engine (IJP), and is controlled by the controller 101. 107 is a network (Ethernet) interface and provides two-way communication which let the interface pass to the controller 101. 108 is an IEEE 1394 interface and provides two-way communication which let the interface pass to the controller 101. 109 is IEEE1284 interface and provides two-way communication which let the interface pass to the controller 101. 110 is a user interface, it comprises an LCD display and a keyboard, and it tells directions from a user to the controller 101 while it performs an information display from the controller 101.

[0014]Multifunctional peripheral equipment which has the above composition makes three physical printers, LBP103 (B/W, with Finisher), LBP104 (B/W), and IJP105 (Color) choose, and enables issue of a print job. Four logic printer (cluster) printers, LBP103+LBP104, LBP104+IJP105, LBP103+IJP105, and LBP103+LBP104+IJP105 are made to choose, and issue of a print job is enabled. The seven above-mentioned printers, Auto select is made from LBP103, LBP104, IJP105, LBP103+LBP104, LBP104+IJP105, LBP103+IJP105, and LBP103+LBP104+IJP105, and issue of a print job is enabled. IJP is used, a logic printer used as a printer which can print only black and white is constituted, and issue of a print job is enabled.

[0015]Issue of the exterior to a scanning job is attained and reading of a color copy becomes possible. A scanner and the eight above-mentioned printers, . [IJP which can print only LBP103, LBP104, IJP105, LBP103+LBP104, LBP104+IJP105, LBP103+IJP105, and LBP103+LBP104+IJP105 black and white] It is made to choose (auto select is also possible), and issue of a copy job is enabled. Only in IJP, a printer enables issue of a color copy job.

[0016]A font and a form overlay can be used by a print job, upload download is possible and resource administration is possible. A color profile can be used by a print job scan job copy job, upload download is possible and resource administration is possible. The automatic generation of the log can be carried out inside apparatus, and upload becomes possible.

[0017]It is usable in all the functions from every interface of Network (Ethernet, TCP/IP), IEEE1284, and IEEE 1394. A subaddress, physics and a logic device (a printer, a scanner, a copy), and correspondence of each resource are taken for every interface, and it becomes possible by choosing a subaddress to choose these physics, logic devices, and each resource. It becomes possible simultaneously to each subaddress of each of these interfaces to direct issue and download upload about a job. A subaddress, physics and a logic device, and correspondence with each resource are obtained by asking apparatus itself. It becomes usable about a function also from a user interface which it has inside.

[0018][by asking Supervisor later mentioned through the above-mentioned interface from the exterior] A provided functional outline (or [what kind of job can be published, / that what kind of resource can be used], etc.), It becomes acquirable [a subaddress used for job issue and download upload of a resource, and detailed information (the number of PDL in which maximum copy number of copies and a finisher are carrying out the kind support, and the outputs BIN which can be specified, etc.) of each function]. It also becomes possible to use the above-mentioned information and to constitute software (control program) from a Client side automatically. However, only the subaddress used for this inquiry needs to know Client beforehand in each interface. Restriction of an administrator is attained by connection form and a use user in the various above-mentioned functions.

[0019](Whole system structure) Drawing 2 is a figure showing system structure of this embodiment. In drawing 2, 201 is multifunctional peripheral equipment (MFT) shown in drawing 1, and the multifunctional peripheral equipment 201 The network interface cable 208 (10 BASE-T), It is connected to ClientPC202, and 203, 204 and 205 (information processor) by the IEEE 1394 interface cable 206 and the IEEE1284 interface cable 207, respectively. In each Client PC as shown in drawing 78, the various software (control program) memorized by DISK6009 later mentioned as a storage shown in drawing 77 is operating. An IP address and a port number are specified, it connects with the multifunctional peripheral equipment 201, and ClientPC202 connected to Ethernet208 and 203 output IP packet data. ClientPC204 connected to the IEEE 1394 interface 206 specifies the nodes ID and LUN (logical unit number), connects with the multifunctional peripheral equipment 201, and outputs SBP-2 packet data. ClientPC205 connected to the IEEE1284 interface 207 specifies a socket number, connects with the multifunctional peripheral equipment 201, and outputs IEEE1284.4 packet data.

[0020](Hardware constitutions of a controller) Drawing 3 is a figure showing hardware constitutions of the controller 101 of this embodiment shown in drawing 1. Inside the controller 101, the LCD display 303 in which CPU301 constitutes Memory(RAM)302 and the user interface (final controlling element) 110 via the bus 313 and the keyboard 304, ROM314, and DISK315 are connected. Various programs and data which are shown in drawing 76 are memorized by DISK315 (storage), such as a hard disk and a floppy disk, are read to Memory(RAM)302 one by one if needed, and are performed by CPU301. This DISK315 could be removable to MFT201, or could be built in MFT201. A program shown in drawing 76 The network interface cable 208 (10 BASE-T), Composition which downloads from ClientPC202, 203, 204 and 205, or other MFT via the IEEE 1394 interface cable 206 and the IEEE1284 interface cable 207, and is memorized by DISK315 may be used.

[0021]The LCD display 303 and the keyboard 304 are what constitutes the user interface (final controlling element) 110 shown in drawing 1, When it displays when CPU301 writes data in the LCD display 303, and CPU301 reads data from the keyboard 304, directions from a user are inputted.

[0022]The network interface connector 305, IEEE 1394 interface connector 306, and IEEE1284 interface connector 307 is connected to the bus 313, It corresponds to the network interface 107 shown in drawing 1, respectively, the IEEE 1394 interface 108, and the IEEE1284 interface 109, The Ethernet (10 BASE-T) cable 208 shown in drawing 2, the IEEE 1394 cable 206, and the IEEE1284 cable 207 are connected. When CPU301 reads or writes in data from these interfaces, communication which uses each interface is performed.

[0023][the bus 313] . [Laser Beam Printer Engine308, Finisher309,Scanner Engine310, Laser Beam Printer Engine311, Ink Jet Printer Engine312] It is connected and, [these] . [drawing 1] . [shown Laser Beam Printer Engine103, Finisher106,Scanner Engine102, Laser Beam Printer Engine104, Ink Jet Printer Engine105] It corresponds, and when reading and CPU301 write data to these engines, engine performances, such as a print scan, and various status acquisition are performed. In addition, [Laser Beam Printer Engine308, Finisher309,Scanner Engine310, Laser Beam Printer Engine311, Ink Jet Printer Engine312] It exists as peripheral equipment of a simple substance not on an inside of MFT201 but on a network, respectively, and the controller 101 of MFT201 may control it.

[0024](Software configuration of the controller 101) Drawing 4 is a figure showing software (control program) composition of DISK315 in the controller 101, and CPU301 performs such software (control program). In a figure, a solid line shows data and control and a dotted line shows a setup and acquisition of capability. 401 is a user interface driver and performs control of the LCD display 303 and the keyboard 304. 405 is a user interface manager (control program), it interprets input from a user inputted from the UI driver 401, generates a command packet shown by drawing 46, and outputs it to the interpreter (control program) 409. The user interface manager 405 interprets a command packet inputted from the interpreter 409 again, and displays on the LCD display 303 via the UI driver (control program) 401.

[0025]402 is a network interface driver (control program), The network interface connector 305 is controlled and a transport packet to processing of a physical layer (physical packet) of a network packet, i.e., extraction of a physical packet to a transport packet, and a physical packet are generated. 406 is a TCP/IP-UDP/IP processing module, it processes a transport packet outputted from the network interface 402, extracts a command packet, and outputs it to the interpreter 409. A command packet to a transport packet outputted from the interpreter 409 is generated, and it outputs to the network interface 402.

[0026]403 is IEEE1284 driver (control program) and controls the IEEE1284 interface 307. 407 is a processing module (control program) of IEEE1284.4. IEEE1284.4 is a standard of a transport layer aiming at mainly being used with IEEE1284 interface. The processing module (control program) 407 of IEEE1284.4 processes a transport packet outputted from the IEEE1284 driver 403, extracts a command packet, and outputs it to the interpreter 409. A command packet to a transport packet outputted from the interpreter 409 is generated, and it outputs to the IEEE1284 driver 403. 404 is an IEEE 1394 driver (control program), and controls the IEEE 1394 interface 306.

[0027]408 is a module (control program) which processes SBP(serial bus protocol)-2 which is a transport layer in IEEE 1394. The processing module 408 of SBP-2 processes a transport packet outputted from the IEEE 1394 driver 404, extracts a command packet, and outputs it to the interpreter 409. A command packet to a transport packet outputted from the interpreter 409 is generated, and it outputs to the IEEE 1394 driver 404. 409 is a packet interpreter which interprets a command packet inputted by each transport treating part and the user interface manager 405 of 406, 407, and 408, and generates a command. The packet interpreter 409 also performs generation of a command packet by the request from other modules (control program). A subaddress in which Supervisor(generalization management manager)410 has the interpreter 409 : [according to a service ID conversion table (drawing 9)] Processing only a command packet inputted from an effective subaddress, a command packet inputted from the other subaddress cancels.

[0028]410 Drawing 7 (attribute table), drawing 9 (a subaddress: service ID conversion table), Drawing 11 (service ID: task type conversion table), drawing 13 (user authentication table), It is Supervisor (generalization management manager) which holds to DISK315 various data shown by drawing 14 (access control table), drawing 16 (event setting table), and drawing 17 (event format table), and carries out generalization management of the operation of the controller 101. Various data in which he holds Supervisor410, Each manager (scanning job manager 419 the font manager 413, the form overlay manager 414, the log manager 415, the color profile manager 416, the print manager 501-509) And according to directions of a command into which it was inputted, reference and change of various data which the copy job manager 601-608 holds, and various data which each controller (the printer controller 510-512, the scanner controller 420) holds are done.

[0029][a command packet which 411 is a security gate and was interpreted by the interpreter 409] [according to a user authentication table (drawing 13) and an access control table (drawing 14) which Supervisor410 (control program) has] Only a user with the right royalty restricts an input of a command so that he may make the multifunctional peripheral equipment 201 usable. [a command which 412 is a dispatcher (control program) and was inputted from the interpreter 409] . [these] Each resource manager who is a control program for processing. (The font manager 413, the form overlay manager 414, the log manager 415, the color profile manager 416), and each job manager (scanning job manager 419 the print manager 501-509) And the copy job manager 601-608 is supplied widely. 413 is a font manager and manages a font. The font manager 413 holds data as shown by drawing 26 (attribute table) and drawing 28 (FONT table) to DISK315. A font (DISK315 or ROM314 memorize) managed by the font manager 413 is passed to the PDL rasterizers 417 and 418, and is used in the case of processing of a print job.

[0030]414 is a form overlay manager and manages a form overlay. The form overlay manager 414 holds data as shown by drawing 29 (attribute table) and drawing 31 (FORM table) to DISK315. A form overlay (DISK315 or ROM314 memorize) managed by the form overlay manager 414 is passed to the PDL rasterizers 417 and 418, and is used in the case of processing of a print job. 415 is a log manager and manages a log. The log manager 415 holds data as shown by drawing 32 (attribute table) and drawing 34 (LOG table) to DISK315. A log is outputted by each manager and put under a log manager's management. A log file of operation is memorized by RAM302 or DISK315, and is updated as required.

[0031]It is a 416 color profile manager and color profile data for a color scanner and a color printer to perform color matching is managed. The color profile manager 416 holds data as shown by drawing 37 (attribute table) and drawing 39 (color profile table) to DISK315. [a color profile managed by the color profile manager 416] Each printer controllers (control program) 510, 511, and 512 and the scanner controller (control program) 420 are passed, and it is used in the case of processing of a print job copy job and a scanning job. 417 is a PDL rasterizer which processes document data which was able to be applied in PS (registered trademark) which is a kind of PDL (page description language), and outputs image data. 418 is a PDL rasterizer which processes document data which was able to be applied in PCL (registered trademark) which is a kind of PDL (page description language), and outputs image data. 419 is a scanning job manager and manages a scanning job. The scanning job manager 419 holds data as shown by drawing 22 (attribute table) to DISK315. The scanning job manager 419 directs execution of a scan for the scanner controller 420. 420 is a scanner controller for controlling the scanner engine 310. The scanner controller 420 holds data showing a function, a state, and performance of the scanner engine 310 as shown by drawing 44 (attribute table) to DISK315. Although the scanning job manager 419 and the scanner controller 420 have data (drawing 22 and drawing 44) in which a function and performance are shown, respectively, their function shown by these data does not necessarily need to be the same. By rewriting data (drawing 22) which the scanning job manager 419 holds through Supervisor410, it is possible to restrict a function.

[0032][the font manager 413, the form overlay manager 414, the log manager 415 and the color profile manager 416 who mentioned above] it is a resource administration program -- the scanning job manager 419, the print job managers 501-509, and the copy job managers 601-608 -- a logic device (a scanner.) It is a printer control program and the laser beam printer controllers 510 and 511, the ink-jet printer controller 512, and the scanner controller 420 are physical device (printer, scanner) control programs.

[0033](A print job manager's composition) Drawing 5 is a figure supplementary to a software (control program) lineblock diagram shown by drawing 4. The dispatcher 412 expresses the same thing as the dispatcher 412 who shows by drawing 4, and distributes a command packet to 509 and 513 from the print job manager (control program) 501. With 501 to 509, 513 is a print job manager and manages a print job. Each print job manager has data as shown by drawing 18 (attribute table) in DISK315, respectively. [each print job manager] In order to process a print job. A printer controller to be used (LBP Controller510, LBP Controller511, Ink Jet Controller512 or Controller510, arbitrary combination of 511 and 512). And a print engine connected to it is decided and it is described by data shown by attribute ID2001 of drawing 18 which printer controller (control program) is used. However, only the print job managers 508 and 513 have a function which chooses a print engine used dynamically, and having this function is described by data (attribute table) which the print job managers 508 and 513 have. . Mean as what kind of job as for each print manager, an inputted print job and these are performed with which printer controller. Data as shown in drawing 20 (job table), drawing 21 (job request table), drawing 90 (job-scheduling table), and drawing 91 (the order job queue table of execution time) is held to DISK 315. 510 to 512 is a printer controller which controls each print engines 308, 311, and 312. The finisher 309 also controls the printer controller 510. Each printer controller has data showing a function, a state, and performance of corresponding printer engine as shown by drawing 40 (attribute table) in DISK315. Each controller holds data as shown in drawing 42 (job queue table) showing a state of an inputted print job to DISK315.

[0034](An interruption print job manager's function) The print job manager 513 of drawing 5 has a special function which other print job managers have, and is called an "interruption print job manager." That is, present each print job manager wedges himself in at least one of the print jobs under an injection / execution to each print controller, and has a function in which a job can be performed previously. A job under interrupted execution will interrupt execution, when all pauses that a print engine allows are good (at for example, the delivery time of each page). In this example, a print engine repeats a single function to only record and discharge a picture on a record paper of an one-sheet unit without a concept of a job, and is only performed.

[0035]Therefore, interruption of a substantial print job is realized by function of a print controller. The interruption print job manager 513 has the function to choose dynamically all the print controllers which exist in a system in consideration of the urgency of a job called interruption, and to make a job perform. One is chosen for a print controller which has what is not performing a job from inside of each print controller, or what has engine quick print speed in fact, and a print job is published with directions of interruption to an interrupt control function which the print controller has. Although data as the interruption print job manager 513 as well as other print managers shows to drawing 90 (job-scheduling table) and drawing 91 (the order job queue table of execution time) is held to DISK 315, A job execution priority and job execution time specification to the print job manager 513 are meaningless on character of the function. Therefore, it is ignored even if it specifies the above-mentioned option (a job execution priority, job execution time) to the interruption print job manager 513.

[0036](Function of a print controller) All print controller (510, 511, 512) is held inside by making a situation (context) of a job under present execution into a variable. For example, a position (bottle ID) of an output bottle of the print engine 308 provided with a finisher unit, a read-out position (file pointer), the remaining print number of sheets of a job file which self holds, etc. correspond. All the print controllers are provided with a field in which these contexts are stored aside from the usual maintenance field in DISK315. When job interruption directions are supervised and a job is supplied with these directions also during job execution, a context of a job under present execution is initialized to the interruption jobs supplied after shunting to said store field. Then, a picture is made to record to a print engine. Interruption directions new during execution of an interruption job are not received. When an interruption job is completed, a job at the time of a context of a shunted job to be interrupted returning and being interrupted is resumed.

[0037]509, 513, and the printer controllers 510-512 have in DISK315 data like drawing 18 (attribute table) in which a function and performance are shown, respectively, or drawing 40 (attribute table) from the print job manager 501. Generally a print job manager has the function to hit a least common denominator of a function which has the performance which is in charge of total of performance which each controller to be used has, and each controller has. However, a part of these functions can be changed by rewriting data (drawing 18) which a print job manager holds through Supervisor410. For example, since the print job manager 503 and the print job manager 509 are using the same printer controller 512, they can provide same performance and function, but. By setting processing which data which a print job manager holds mentions later, the print job manager 503 can color-print and can also make impossible color printing of the print job manager 509. It is set up whether a job management setup (a job execution priority, job execution time, job residual limit time) is possible for each print job manager other than interruption print job manager 513 at attribute ID2006. This attribute is set as YES by default.

[0038](A copy job manager's composition) Drawing 6 is a figure supplementary to a software (control program) lineblock diagram shown by drawing 4 and drawing 5. The dispatcher 412 expresses the same thing as the dispatcher 412 to whom it is shown by drawing 4, and distributes a command packet to the copy job managers (control program) 601-609. 601 to 609 is a copy job manager, and manages a copy job. Each copy job manager has data as shown by drawing 24 (attribute table) in DISK315, respectively. A printer controller used in order that each copy job manager may process a copy job, A print engine and a scanner engine which were connected to a scanner controller and it are decided, and it is described by data (attribute ID2001) shown by drawing 24 which printer controller and scanner controller are used. However, only the copy job managers 608 and 609 have a function which chooses a print engine used dynamically, and having this function is described by data (attribute ID2002) which the copy job managers 608 and 609 have. Although ****** is later mentioned in detail to drawing 24, It is set up like drawing 18 and drawing 19 which were mentioned above whether a job management setup (a job execution priority, job execution time, job residual limit time) is possible for each copy job manager other than interruption copy job manager 609 at attribute ID2006. This attribute is set as YES by default.

[0039]An inputted copy job and these each copy manager Which printer controller, Data as shown in drawing 20 (job table), drawing 21 (job request table), drawing 90 (job-scheduling table), and drawing 91 (the order job queue table of execution time) which mean as what kind of job it performs in a scanner controller is held to DISK315. 510 to 512 is a printer controller which controls each print engine, and expresses the same thing as what is shown in drawing 5. Although it differs from a print job manager in that a copy job manager makes a manuscript read using a scanner controller, and image data is received, a procedure which supplies a job to a subsequent print controller is completely the same.

[0040](An interruption copy job manager's function) The copy job manager 609 of drawing 6 has a special function which other copy job managers have, and is called an "interruption copy job manager." That is, present each copy job manager wedges himself in at least one of the print jobs under an injection / execution to each print controller, and has a function in which a job can be performed previously. A copy job manager performs a copy job in two steps of supplying a job to a print controller, after supplying a job to a scanning controller. It interrupts in this example, and a copy job is not performed but interrupts only a job under injection to a print controller below interrupting a job supplied to said scanning controller. That is, interruption processing is made to perform using other printer engine without using printer engine which was planning image data read with a scanning engine at the beginning.

[0041](Attribute table of Supervisor) Drawing 7 expresses data (attribute table) which Supervisor (control program)410 holds and manages to DISK315. This table expresses a functional outline, connection information, security information of the multifunctional peripheral equipment 201, etc. In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records. Each record comprises attribute ID701, model ID702, and the attribute value 703, and expresses an attribute which Supervisor has. Attribute ID701 expresses a kind of information and it is shown what kind of meaning what the value 703 shows by this has. Attribute ID701 is unique inside apparatus and a thing with the same attribute ID expresses a kind of the same information. Model ID702 shows what kind of data type the value 703 has, and when it interprets the value 703, it is used. According to the attribute ID, the model ID is decided to be a meaning and is uniquely defined by inside of apparatus. Although put into both attribute ID701 and model ID702 in an attribute table in this embodiment, a conversion table of the attribute ID and the model ID may be held as data with an another attribute table, and it may carry out so that only the attribute ID and a value may be put in an attribute table. The value 703 shows an attribute value according to attribute ID701.

[0042]Drawing 8 is a figure showing the details of attribute ID701 of the attribute table of drawing 7, and model ID702. In this, the "list of the security level currently supported" of attribute ID102 holds the list of security levels which can be set up about the security level needed when publishing operation to Supervisor. About a security level, it is made in explanation of the access control table of drawing 14. The "present security level" of attribute ID103 expresses the security level set as Supervisor now. The "present count data" of attribute ID401 expresses the list of information used as the candidate for a count in Supervisor, and this meaning place is denoted by the "count data format" of attribute ID402. For example, the count data 45, 78, 34, and 13 shows the print number of sheets of paper size denoted by the count data format. The "count data format" of attribute ID402 is expressed as a list of attributes ID. Since the attribute ID is uniquely defined within apparatus, it can specify the meaning of count data by specifying the attribute ID. For example, in the attribute value 565 of attribute ID402, the print number of sheets of the paper size A2 and the attribute value 537 mean the print number of sheets of the paper size A3, the attribute value 545 means the print number of sheets of the paper size A4, and the attribute value 523 means the print number of sheets of the paper size A5. The count restriction data of attribute ID403 expresses the maximum which the "present count data" of attribute ID401 can have, The subaddress CPU301 is indicated to be by drawing 9 when at least one of the values of "the present count data" of this maximum is exceeded: Repeal all the (based on a manager) services by repealing the effective flag of a service ID conversion table. The count unit price data of attribute ID404 expresses with the currency unit the unit price for every count of the attribute ID currently held at the "count data format" of attribute ID402. The present billing data of attribute ID405 shows the value which multiplied the value to which the "present count data" of attribute ID401 and the "count unit price data" of attribute ID404 correspond, and took total. The fee collection restriction data of attribute ID406 expresses the maximum which can have the "present billing data" of attribute ID405, The subaddress CPU301 is indicated to be by drawing 9 when "the present billing data" of attribute ID405 exceeds "fee-collection restriction data" of attribute ID406: Repeal all the (based on a manager) services by making the effective flag of a service ID conversion table into an invalid flag. The event list which is supporting attribute ID501 expresses the event which can set up what is notified to Supervisor with the list of events ID defined as a meaning for every kind of event. A setup of actual event transmission is performed by setting the connection form and the address of an event notification to the event setting table shown by drawing 16 with the event ID. It is shown that what is described to be "Table Data" about the value of the attribute table of drawing 7 is data in which a value has the tabular format (52-104) shown with the model ID.

[0043](A subaddress: Service ID conversion table) drawing 9 -- subaddress: -- a service ID conversion table is expressed. This table is held DISK315 as an attribute value of attribute ID1001 (a subaddress: SID conversion table) of an attribute table shown by drawing 7. A subaddress: Which services (for example, a print, download of a font, etc.) are obtained, and a SID conversion table shows **, if a command packet is supplied to which subaddress. It is controlled whether according to an effective flag of this table, etc., the interpreter 409 provides job service for every connection form. In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records. Each record comprises the list 805 of connection formal ID801, the subaddress 802, service ID803, the effective flag 804, and effective users, and the list 806 of invalid users. Connection formal ID801 shows an identifier showing a connection form. Drawing 10 is a figure showing a meaning of the connection form ID, as for 0, Internal (user interface) and 1 express TCP/IP (network), two express IEEE1284.4 (IEEE1284), and 3 expresses SBP-2 (IEEE 1394).

[0044]The subaddress 802 expresses a subaddress for every connection form. Although a user interface does not have a subaddress, it is assigning a subaddress for convenience and supplies a command packet from a user interface with this subaddress information.

[0045]Service ID803 is an identifier showing service. Service ID803 is assigned to each manager to whom it was shown by drawing 4, drawing 5, and drawing 6 by 1 to 1.

[0046]It is expressed whether the subaddress of the effective flag 804 is effective. When this value is truth (YHS), a subaddress is effective, and a user can publish a command packet to a subaddress. It means that a packet is canceled, even if a subaddress is invalid and it publishes a command packet to a subaddress, when a value is an imitation (NO). In the case of drawing 8, when a connection form of connection formal ID2 (IEEE1284) is used, it means that service of service ID8 (print job manager) cannot be provided.

[0047]The list 805 of effective users expresses a list of use IDs which can receive the service, when a subaddress is effective. The list 806 of invalid users expresses a list of use IDs which cannot receive the service, even when a subaddress is effective. The list 805 of effective users and the list 806 of invalid users can set a value only to either.

[0048](Service ID: Task type conversion table) Drawing 11 expresses a service ID:task type conversion table. This table is held DISK315 as an attribute value of attribute ID1002 (SID : Task Type table) of an attribute table shown by drawing 7. Service ID: A task type table means what kind of service the service ID provides. In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records. Each record comprises service ID901 and the task type 902. It is shown what kind of service the task type 902 is. Drawing 12 is a figure showing a meaning of a task type 902 value. A print and 2 interrupt below service which Supervisor offers 0, and 1, a copy and 11 interrupt below a scan and 3, a print and 13 interrupt, a font and 102 express a form overlay, 103 expresses a log, and a copy and 101 express a color profile 104. The service ID corresponds by 1 to 1 with a manager who provides service. For this reason, the service ID is used also for access of a functional table which each manager has. Since a functional table also has each printer controller and a scanner controller, [a table] In order to access these, the controller ID equivalent to the service ID is also assigned to a printer controller and a scanner controller, and it is managed by a service ID:task type conversion table what kind of controller the controller ID is. The task type 201 expresses a printer controller and 202 expresses a scanner controller. The controller ID corresponds by each controller shown by drawing 4 and drawing 5, and 1 to 1.

[0049]The meaning of the service ID is as follows.

0 : Supervisor4101 : . Print job manager 5012 : Print job manager 5023: Print job manager 5034 : Print job manager 5045 : Print job manager 5056 : The print job manager 5067. : Print job manager 5078 : . Print job manager 5089 : Print job manager 50910 : Scanning job manager 41911 : Copy job manager 60112 : Copy job manager 60213 : The copy job manager 60314. : Copy job manager 60415 : . Copy job manager 60516 : Copy job manager 60617 : Copy job manager 60718 : The copy job manager 60819 : The interruption print job manager 51320 : The interruption copy job manager 609101. : Font manager 413102 : Form overlay manager 414103 : Log manager 415104 : Color profile manager 41621 : Printer KONTORO. - RA 51022: Printer controller 51123 : Printer controller 51224 : The scanner controller 420 [0050](User authentication table) Drawing 13 expresses a user authentication table which Supervisor holds, and expresses attestation information of a user who can use apparatus. This table is held DISK315 as an attribute value of attribute ID1003 (user authentication table) of an attribute table shown by drawing 7. A user authentication table expresses a group of an effective use ID and a password with information on whether the user has an administrator privilege. In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records. Each record comprises the use ID 1001, the password 1002, and the administrator privilege flag 1003. The administrator privilege flag 1003 means whether a user has a right as an administrator.

[0051](Access control table) Drawing 14 expresses an access control table which Supervisor holds, and expresses a security level for every service. This table is held DISK315 as an attribute value of attribute ID1004 (access control table) of an attribute table shown by drawing 7. An access control table holds a list of users allowed a required security level and service offer for every service ID. In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records.

[0052]Each record comprises the list 1103 of service ID1101, the security level 1102, and use IDs. When receiving service specified by service ID1101 that is, the security level 1102 expresses required attestation information, when it publishes a command packet to a manager to whom it is shown by service ID1101. Drawing 15 is a figure showing a meaning of a security level. Although only not needing user authentication, only an administrator's distinguishing 1, and a user of the security level 0 with whom 2 was attested are usable, Only as for an attested user, not using a password for attestation using a use ID and 3 are usable, and it means using a use ID and a password for attestation. The list 1103 of use IDs expresses a list of use IDs to which use (access) was permitted, when the security levels 1102 are 2 and 3.

[0053](Event setting table) Drawing 16 expresses an event setting table which Supervisor holds. This table is held DISK315 as an attribute value of attribute ID502 (event setting table) of an attribute table shown by drawing 7. An event setting table holds a method and an address which transmit an event notification for every kind of event, when a specified event occurs within apparatus. In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records. Each record comprises event ID1201, the connection form 1202, and the notice place address 1203. An address comprises a notice place address depending on connection form and connection form. Event ID1201 expresses a kind of event and is uniquely defined within apparatus. For example, toner nothing and ID432 show ink nothing and, as for paper nothing and ID399, Event ID200 shows cover opening of multifunctional peripheral equipment ID234. Connection formal ID1202 expresses connection form which transmits an event notification, and the thing as what is used by a subaddress:service ID conversion table of drawing 9 with this same value is used. The notice place address 1203 expresses a notice place address of an event which suited the connection form 1202.

[0054](Event format table) Drawing 17 expresses an event format table which Supervisor holds, and shows the contents of the event. This table is held DISK315 as an attribute value of attribute ID503 (event format table) of an attribute table shown by drawing 7. About each event ID uniquely defined within apparatus, an event format table holds form of addition data transmitted as an event notification. In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records. Each record comprises event ID1301 and the event format 1302. The event format 1302 is the form of addition data transmitted as an event notification, and is expressed as a list of attributes ID. The attribute ID is uniquely defined within apparatus, and since the model is also determined as a meaning by the attribute ID, it can express a format of addition data by specifying the attribute ID. For example, attribute ID676 of the event format 1302, a kind of toner and ID698 show a kind of ink and, as for paper size and ID756, ID600 shows a position of a cover, as for a kind of paper, and ID666. As shown below, it is possible to notify an event which a transmission method and an address of an event are set as data (attribute table) which each manager and a controller hold, and occurred within each of these managers and a controller, but. A format of addition data of these events is also held in an event format table shown by drawing 13. When a specified event occurs, contents defined by event format are notified with data beforehand decided for every event ID.

[0055](A print job manager's attribute table) Drawing 18 expresses data (attribute table) which 509 and 513 hold from each print job manager (control program) 501 to DISK315. This table expresses performance and a function of a print job which can be treated by print job manager. In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records. The contents of each record are the same as that of a thing of Supervisor shown by drawing 7, and comprise attribute ID1401, model ID1402, and the value 1403. Drawing 19 is a figure showing a value of attribute ID1401 of an attribute table of drawing 18, and model ID1402. Model ID203 (job tabular format) of an attribute table of drawing 19, model ID204 (job request tabular format), Model ID205 (job-scheduling tabular format), model ID81 (prohibition attribute combination list 1 form), Model ID82 (prohibition attribute combination list 2 form), model ID83 (prohibition attribute combination list 3 form), About other models ID other than model ID84 (prohibition attribute combination list 4 form) and model ID85 (prohibition attribute combination list 5 form), it is the same as that of what is used in an attribute table of Supervisor of drawing 7.

[0056]["a downloading method of data currently supported" of attribute ID601] It is a thing showing a transmission method to apparatus of document data to print, this embodiment -- 1: -- a method of including document data into a job. And 2: Inside of a job includes a reference pointer (URL: Unified Resource Locator) of document data, and a method of reading document data shown with a reference pointer if needed from apparatus is supported.

[0057]Attribute ID801, and 802, 803, 804 and 805 are the prohibition setting attributes showing restriction about an attribute set up into a print job supplied to a print job manager. When copy number of copies exceeds 100, for example, since limitations that a setup of a finisher becomes impossible etc. are expressed, these are used. There are five models in a prohibition setting attribute as a mode of expression of the restriction. Attribute ID801 holds two or more pairs of a list of another attribute ID2 to certain attribute ID1, as {attribute ID1:{attribute ID2}} shows. This means that each attribute ID2 cannot be set up in a print job script, when a setup of attribute ID1 is made.

[0058]Attribute ID802 holds two or more pairs of a list of another attribute ID2 to a certain attribute ID1, an operator identifier, and a value, as {attribute ID1:operator identifier:value:{attribute ID2}} shows. This means that each attribute ID2 cannot be set up in a print job script, when a result to which an operator shown by an operator identifier to attribute ID1 is applied to a value is truth. As for an operator identifier, an integral value corresponds by 1 to 1 to 2 paragraph operation. For example, 0 expresses "= (equal)."

[0059]attribute ID803 -- {-- as attribute ID1:operator identifier 1:value 1: {attribute ID2:operator identifier 2:value 2}} shows, two or more pairs of a list of groups of attribute ID2 [another], the operation identifier 2, and the value 2 are held to a certain attribute ID1, the operator identifier 1, and the value 1. When a result to which an operator this is indicated to be by the operator identifier 1 to attribute ID1 is applied to the value 1 is truth, it means that a setup that a result to which an operator shown operator identifier 2 to each attribute ID2 is applied to the value 2 becomes truly cannot be performed in a print job script.

[0060]As {(the attribute ID1:operator identifier 1: value 1) :operation identifier 3:(attribute ID2:operator identifier 2: value 2):{attribute ID3}} shows, [attribute ID804] Two or more pairs of a list of another attribute ID3 are held to attribute ID2 [different from a certain attribute ID1, the operator identifier 1, and the value 1 and the operation identifier 3], the operation identifier 2, and the value 2. When a result of having done an operation shown by the operator identifier 3 in a result to which an operator shown by the operator identifier 2 to a result and attribute ID2 to which an operator shown by the operator identifier 1 to attribute ID1 is applied to the value 1 is applied to the value 2 is truth, [this] It means that a setup to each attribute ID3 cannot be performed in a print job script.

[0061]attribute ID805 -- {(the attribute ID1:operator identifier 1: value 1), [that :operation identifier 3: (attribute ID2:operator identifier 2: value 2): {attribute ID4:operation identifier 4:value 4}} shows] Two or more pairs of a list of groups of attribute ID4 [another], the operation identifier 4, and the value 4 are held to attribute ID2 [different from a certain attribute ID1, the operator identifier 1, and the value 1 and the operation identifier 3], the operation identifier 2, and the value 2. When a result of having done an operation shown by the operator identifier 3 in a result to which an operator shown by the operator identifier 2 to a result and attribute ID2 to which an operator shown by the operator identifier 1 to attribute ID1 is applied to the value 1 is applied to the value 2 is truth, [this] It means that a setup that a result to which an operation which calculates to each attribute ID4 and is shown by 4 is applied to the value 4 becomes new cannot be performed in a print job script.

[0062]["a controller (it is possible) ID list which performs Job" of attribute ID2001] It is a list of controllers ID of a controller which performs a print job, and print job managers other than print job manager 508 have become a list of controllers which certainly perform a job. In order that the print job manager 508 may choose a controller dynamically by the function in which a job is demanding a controller, this attribute expresses a range which chooses a controller. "controller of attribute ID2002 -- auto select -- " -- it is meant whether a print job manager makes auto select of the controller. According to this embodiment, only the print job manager 508 makes this value truth. About other attributes ID, it is the same as that of what is used in an attribute table of Supervisor of drawing 7. However, a range which a range and influence of a setup attain to is limited to a thing under management of a print job manager to whom an attribute table belongs. [for example "operation currently supported" of the attribute value 101] It is a list of operation which can be published to this print job manager, and "fee collection restriction data" of the attribute value 406 adds restriction of fee collection about a print job performed by this print job manager. Other managers' service is not affected only by handling at the time of exceeding restriction of fee collection repealing this print job manager's service. The service ID of an attribute table of a print job manager who shows drawing 18 is an attribute table (print job manager 504) of four, An attribute table of service ID1 and the print job managers 501, 502, 503, 505, 506, 507, 508, 509, and 513 corresponding to each of 2, 3, 5, 6, 7, 8, 9, and 13 is memorized by DISK315, and differ, respectively.

[0063](Job table) Drawing 20 is a figure showing data (job table) which each print job manager holds, and shows a conversion table of a filename and the job ID holding substance of a job which a print manager manages. A table is held RAM302 as an attribute value of attribute ID2003 (job table) of an attribute table shown by drawing 18. Each print job manager changes a job table on RAM302 dynamically, and saves DISK315 if needed. In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records. Each record comprises the filename 1502, the execution priority 1506, the specification execution time 1507, and the residual limit time 1508 when substance of job ID1501 and a job is held. Job ID1501 is an identifier to a job which a print job manager assigned, when a job packet is supplied from Client PC (202-205) and a job is supplied to a print job manager. The filename 1502 of a job is a name of a file by which substance of a job is held. Substance of a job is constituted by having two or more groups of the attribute ID, attribute value size, and an attribute value, as shown in drawing 52.

[0064]Although mentioned later for details, [the execution priority 1506, the specification execution time 1507, and the residual limit time 1508] When a job is supplied to a print job manager, it is an attribute simultaneously specified based on specification information included in a job packet from Client PC (202-205), and the specification value is altogether stored in a job table shown in drawing 20. The execution priority 1506 has a natural number with a range as the contents. By this example, a priority takes the range of 1 to 3, and this value (priority) means that an execution priority of a job is so high that it is small. This value can be specified as 2 thru/or 1 to perform [client / of a job] that job early comparatively. If there is no specification, the default priority 3 is given automatically and stored in this job table. A job is registered into a job-scheduling table later mentioned according to an execution priority, and is sent to a controller related to order with a high priority. After this priority supplies a job, it can be changed via Supervisor. A value of a priority of a job table is changed in that case, and execution scheduling of a job which a scheduling table was updated immediately and followed a new priority is performed.

[0065]The specification execution time 1507 is specified when execution time of a job needs to be specified. If there is no specification, execution of a job will be processed based on the above-mentioned priority. When specification execution time is specified, it is not based on specification of an execution priority, but a priority is compulsorily set as 1 (the minimum priority). Specification execution time needs to be set up with future than time when a job was supplied. A job which had specification execution time specified is not immediately registered into a job-scheduling table at the job injection time. It registers with the order job queue table of execution time instead. When it reaches at specification execution time, registration is deleted from the order job queue table of execution time, and it registers with a job-scheduling table with the priority 1 immediately. That is, a supplied job will certainly be registered into order job queue either one of each job manager's job-scheduling table or a table of execution time. This specification execution time can specify 11:00 a.m. with a user interface of Client PC (202-205) as time absolutely to perform a job at 11:00 a.m., and, for example, [00] [the time] [at present 10:00 a.m.] From now on, less than 30 minutes can be specified from a user interface of Client PC (202-205) as relative time to perform a job within 30 minutes from now on. The former is effective to perform a job at a case where you want to perform a lot of jobs, for example in consideration of energy saving at night, or lunch break time, The latter is effective to perform a job by one's time sensation in a time schedule of work currently drawn into the head, without taking time into consideration now. And when specification execution time is specified at relative time, when a job is supplied to each job manager, CPU301 calculates time absolutely from relative time specified as the present time at that time, and it registers with a job table of drawing 20, and the order job queue table of execution time of drawing 84.

[0066]Residual limit time 1508 attribute sets the survival term in the apparatus of a job. As for the usual job, execution may be delayed according to the Assessment on Search Report by Designated

Searching Authority of an execution priority, execution start time, and the controller used. It has become [being stored with as in the state of un-performing for a while in apparatus, and] in that case. However, there is a thing not to neglect by un-performing in apparatus too much from the importance and airtight height for a long period of time depending on a job. In this example, since such a demand is satisfied, the residual limit time of a job can be set up. That is, the job to which this attribute was set cannot exist by a non-run state in apparatus more than this specified time, but is deleted automatically. Residual limit time is used together with previous execution start time, and can be set up. Although it is more desirable than execution start time to specify the time of the future as for residual limit time, when the condition is not fulfilled, the job will be deleted immediately after the injection of a job. This residual limit time can specify 11:00 a.m. with the user interface of Client PC (202-205) as time absolutely, for example to delete a job within the limit of 11:00 a.m. at present 10:00 a.m., and, [00] [the time] From now on, less than 30 minutes can be specified from the user interface of Client PC (202-205) as relative time to delete a job within the limit of less than 30 minutes from now on. The former is effective and to delete the job by making into limit time time which a lunch break finishes when an airtight high job cannot be performed during a lunch break for example, [the latter] For example, it is effective to delete the job by making less than 30 minutes into limit time from now on in order to come out of a company in the time schedule of the work currently drawn into the head for going home. And when residual limit time is specified at relative time, when a job is supplied to each job manager, CPU301 calculates time absolutely from the relative time specified as the present time at that time, and it registers with the job table of drawing 20, and the order job queue table of execution time of drawing 84.

[0067](Job request table) Drawing 21 shows relation between a job which is a figure showing a job request table which a print job manager holds, and a print manager manages, and a job performed by a controller. This table is held DISK315 as an attribute value of attribute ID2004 (job request table) of an attribute table shown by drawing 18. A job request table means as which job a job which a print job manager manages is performed by which controller. In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records. Each record comprises job ID1503, controller ID1504, and job ID1505 that are assigned in a controller. When a job is supplied to a print job manager, job ID1503 is an identifier to a job which a print job manager assigned, and corresponds with job ID1501 in a job table (drawing 20). Controller ID1504 expresses ID of a controller by which a job is performed. Job ID1505 is an identifier of a job which a controller which performs a job assigned. Although two or more records which had the same value in job ID1503 in a job table of drawing 21 exist, this shows a case where divide one certain job into two or more controllers, and it is performed. For example, it is shown that a job of job ID1503=1 in drawing 21 was divided as a job to two controllers, controller ID=21 and 22, and job ID1505=100 and 101 were assigned from a controller, respectively.

[0068](Job-scheduling table) Drawing 83 is a figure showing a job-scheduling table which all the print job managers and all the copy job managers hold. Each line of a job-scheduling table serves as a list of jobs ID in every job execution priority. That is, the first line of a table is filled in from the left in order of the time when the job ID of the job execution priority 1 was supplied (job ID5, 7, 8). The job group of the execution priorities 2 and 3 of the 2nd line and the 3rd line is filled in similarly, respectively. A job-scheduling table changes every moment according to an injection of a job, and an execution situation. An execution priority is the highest, and a job in a job-scheduling table is taken out from the youngest thing of the order of an injection by job manager (deleted from a table), and is supplied to a ** controller.

[0069](The order job queue table of execution time) Drawing 84 is a figure showing the order job queue table of execution time which all the print job managers and all the copy job managers hold. In the order job queue table of execution time, each line has accomplished one record which consists of absolute time when a job should be performed, and the job ID, and a table is constituted as a set of two or more records. The execution time 1561 and job ID1562 are the same models as the specification execution time 1507 of a job request table of drawing 21, and job ID1503, respectively. A job by which a value is set to the specification execution time 1507 of a job table of drawing 21 is sorted by young order of the specification execution time, and the job ID and execution time are registered into the order job queue table of execution time. That by which execution time specification was made among jobs supplied to a print job manager and a copy job manager is once registered into the order job queue table of execution time, and that at which execution time arrived is deleted from the order job queue table of execution time after that, and it is registered into a job-scheduling table. This specification execution time is specified at relative time or absolute time, as mentioned above, but. When specification execution time is specified at relative time, when a job is supplied to each job manager, CPU301 calculates time absolutely from relative time specified as the present time at that time, and it registers with a job table of drawing 20, and the order job queue table of execution time of drawing 84.

[0070](A scanning job manager's attribute table) Drawing 22 expresses an attribute table which is data which the scanning job manager 419 holds. This table expresses performance and a function of a scanning job which can be treated by scanning job manager. In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records. The contents of each record are the same as that of a thing of Supervisor shown by drawing 7, and comprise attribute ID1601, model ID1602, and the value 1603. Drawing 23 is a figure showing a value of attribute ID 1601 of an attribute table of drawing 22, and model ID1602. ["an uploading method of data currently supported" of attribute ID=602] It is a thing showing a transmission method to apparatus of scanned document data, this embodiment -- 1: -- a method of including document data during a reply of a job. And 2 : Hold document data in apparatus and a reference pointer (URL: Unified Resource Locator) to document data is included during a reply of a job, A method of reading document data shown with a reference pointer if needed from a host (Client) is supported. "A list of a graphics format in which an output is possible" of attribute ID=1201 expresses a data format of a picture in which an output is possible, and when publishing a scanning job, one of the things of a certain can be specified as a data format in this. It is the same as that of what a print job manager to whom a job table of attribute ID=2003 was shown by drawing 20 holds. About other attributes ID, it is the same as that of what is used in an attribute table of Supervisor of drawing 7. However, a range which a range and influence of a setup attain to is limited to a thing under management of a scanning job manager to whom an attribute table belongs. About other models ID other than model ID203 (job tabular format) of an attribute table of drawing 23, it is the same as that of what is used in an attribute table of Supervisor of drawing 7.

[0071](A copy job manager's attribute table) Drawing 24 expresses data (attribute table) which each copy job managers 601-609 hold. This table expresses performance and a function of a copy job which can be treated by copy job manager. In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records. The contents of each record are the same as that of a thing of Supervisor shown by drawing 7, and comprise attribute ID1701, model ID1702, and the value 1703. Drawing 25 shows a value of attribute ID1701 of an attribute table of drawing 24, and model ID1702. Attribute ID1302 (can it color-print?), attribute ID1303 (kind of finishing currently supported), Attribute ID1304 (the highest resolution which can be set up), attribute ID1305 (the minimum resolution which can be set up), attribute ID2001 (controller ID list which may perform Job), attribute ID2002 (a controller -- do auto select or not?), and attribute ID2003 (job table), [** and] Are the same as that of what was shown in explanation of a print job manager and a scanning job manager, Attribute ID2004 (Job request table), attribute ID2005 (Job calendar), and attribute ID2006 (is a job management setup possible or not?) are the same as that of what was shown in a print job manager's explanation. Attribute ID1302 (can it color-print?), attribute ID1303 (kind of finishing currently supported), Attribute ID1304 (the highest resolution which can be set up), attribute ID1305 (the minimum resolution which can be set up), attribute ID2002 (a controller -- do auto select or not?) and attribute ID2003 (job table). Attribute ID2004 (job request table), attribute ID2005 (job calendar), Attribute ID2006 (is a job management setup possible or not?), model ID203 (job tabular format), About the attributes ID and the models ID other than model ID204 (job request tabular format) and model ID205 (job calendar form), it is the same as that of what is used in an attribute table of Supervisor of drawing 7. The service ID of an attribute table of a copy job manager who shows drawing 24 is an attribute table (copy job manager 608) of 18, An attribute table of service ID11 and the copy job managers 601, 602, 603, 604, 605, 606, 607, and 609 corresponding to each of 12, 13, 14, 15, 16, 17, and 19 is memorized by DISK315, respectively, and differ, respectively.

[0072](A font manager's attribute table) Drawing 26 expresses data (attribute table) which the font manager 413 holds to DISK315. This table expresses a type of a font which can be treated by font manager, a list of fonts which the present font manager has managed, etc. In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records. The contents of each record are the same as that of a thing of Supervisor shown by drawing 7, and comprise attribute ID1801, model ID1802, and the value 1803. Drawing 27 shows a value of attribute ID1801 of an attribute table of drawing 26, and model ID1802. Attribute ID601 (downloading method of data currently supported), attribute ID602 (uploading method of data currently supported), Attribute ID1501 (list of font types currently supported), attribute ID1502 (the number of fonts which can be maximum held), The attributes ID and the models ID other than attribute ID1503 (the number of fonts held now), attribute ID1504 (list of fonts currently held), and model ID150 (font table) are the same as that of what is used in an attribute table of Supervisor of drawing 7. However, a range which a range and influence of a setup attain to is limited to a thing under management of a font manager to whom an attribute table belongs. "A downloading method of data currently supported" of attribute ID601, And "an uploading method of data currently supported" of attribute ID602 is the same as that of what is during a print job manager's explanation and a scanning job manager's explanation, and is explained respectively, A downloading method and an uploading method of font data which are supported are shown.

[0073](Font table) Drawing 28 expresses a font table which a font manager holds. This table is held DISK315 as an attribute value of model ID1504 (font table) of an attribute table shown by drawing 26. A font table means what kind of font a font manager has managed now. In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records. Each record comprises the filename 1904 containing font ID1901, the font type 1902, the font name 1903, and font data. Font data is memorized by DISK315, and font ID1901 is an identifier to a font which a font manager assigned, when font data downloads to RAM302.

[0074](A form overlay manager's attribute table) Drawing 29 expresses data (attribute table) which the form overlay manager 414 holds to DISK315. This table expresses a format of a form overlay which can be treated by form overlay manager, a list of form overlays which the present form overlay manager has managed, etc. In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records. The contents of each record are the same as that of a thing of Supervisor shown by drawing 7, and comprise attribute ID2001, model ID2002, and the value 2003. Drawing 30 shows the attribute ID of an attribute table of drawing 29, and a value of the model ID. Attribute ID601 (downloading method of data currently supported), attribute ID602 (uploading method of data currently supported), Attribute ID1601 (list of formats of a form overlay currently supported), Attribute ID1602 (the number of form overlays which can be maximum held), attribute ID1603 (the number of form overlays held now), About the attributes ID and the models ID other than attribute ID1604 (list of form overlays currently held), and model ID160 (form table), it is the same as that of what is used in an attribute table of Supervisor of drawing 7. However, a range which a range and influence of a setup attain to is limited to a thing under management of a form overlay manager to whom an attribute table belongs. "A downloading method of data currently supported" of attribute ID601, And "an uploading method of data currently supported" of attribute ID602 is the same as that of what is during a print job manager's explanation and a scanning job manager's explanation, and is explained respectively, A downloading method and an uploading method of form overlay data which are supported are shown.

[0075](Form overlay table) Drawing 31 expresses a form overlay table which the form overlay manager 414 holds. This table is held DISK315 as an attribute value of attribute ID1604 (form overlay table) of an attribute table shown by drawing 29. A form overlay table means what kind of form overlay a form overlay manager has managed now. In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records. Each record comprises the filename 2104 containing form overlay ID2101, the data format 2102 of a form overlay, the form overlay name 2103, and form overlay data. Form overlay data are memorized by DISK315, and form overlay ID2101 is an identifier to a font which a form overlay manager assigned, when form overlay data download to RAM302.

[0076](A log manager's attribute table) Drawing 32 expresses the data (attribute table) which the log manager 415 holds to DISK315. This table expresses the list of logs which the present log manager has managed, etc. In the table, each line expresses one information unit (record), and data is constituted as a set of two or more records. The contents of each record are the same as that of the thing of Supervisor shown by drawing 7, and comprise attribute ID2201, model ID2202, and the value 2203. Drawing 33 shows the attribute ID of the attribute table of drawing 32, and the value of the model ID. Attribute ID602 (uploading method of the data currently supported), attribute ID1703 (Log number currently held), About the attributes ID and the models ID other than attribute ID1704 (log table), attribute ID1705 (format table of Log), model ID170 (log table), and model ID171 (log format tabular format), it is the same as that of what is used in the attribute table of Supervisor of drawing 7. However, the range which the range and influence of a setup attain to is limited to the thing under management of the log manager to whom the attribute table belongs.

[0077](Log table) Drawing 34 expresses a log table which the log manager 415 holds. This table is held DISK315 as an attribute value of attribute ID1704 (log table) of an attribute table shown by drawing 32. A log table means what kind of log a log manager has managed now. In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records. Each record comprises log ID2301 and the filename 2302 containing log data. Log ID2301 is an identifier of a log beforehand specified to a kind.

[0078](The contents of log data) Drawing 35 expresses the contents of log data which a log manager manages by DISK315.Log data are constituted as a set of a log record. Each log record comprises log format ID2401, the record time 2402, and the log data 2403. Log format ID2401 is referring to format information on a log currently recorded on a log format table shown by drawing 36, and expresses a format of the log data 2403. [for example log data "12345, 4, 1.23, "O.K."" of LogFormatID1] In the amount of toner which the number of papers which a use ID which published a job outputted by 12345 used by 4, 1.23 and an end state of a job show a print job manager of service ID1 O.K.

[0079](Log format table) Drawing 36 expresses a log format table which the log manager 415 holds. This table is held DISK315 as an attribute value of the attribute ID (log format table) of an attribute table shown by drawing 32. A log format table expresses a format of a log currently recorded within each log data which the log manager 415 manages. In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records. Each record comprises log format ID2501 and the log format 2502. A log format is expressed as a list of service ID:attribute ID pairs. The attribute ID is uniquely defined within apparatus, and since the model is also determined as a meaning by the attribute ID, it can express a format of log data by specifying the attribute ID. Service ID of the log format 2502: A list of attribute ID pairs and a list of values in the log data 2403 correspond by turn. For example, 3rd service ID of log format 2502 list: An attribute ID pair means a format of the 3rd value of log data. For example, a format "1:701, 1:565, 1:765, 1:777" of LogFormatID1 shows a print job manager of service ID1 a use ID which published a job, the outputted number of papers, the used amount of toner, and an end state of a job. A format "11:701, 11:565, 11:765, 11:777" of LogFormatID2 shows a print job manager of service ID11 a use ID which published a job, the outputted number of papers, the used amount of toner, and an end state of a job.

[0080](A color profile manager's attribute table) Drawing 37 expresses data (attribute table) which the color profile manager 416 holds to DISK315. This table expresses a format of a color profile which can be treated by color profile manager, a list of color profiles which the present color profile manager has managed, etc. In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records. The contents of each record are the same as that of a thing of Supervisor shown by drawing 7, and comprise attribute ID2601, model ID2602, and the value 2603. Drawing 38 shows the attribute ID of an attribute table of drawing 37, and a value of the model ID. Attribute ID601 (downloading method of data currently supported), attribute ID602 (uploading method of data currently supported), Attribute ID1801 (list of formats of color profile data currently supported), Attribute ID1802 (the number of color profiles which can be maximum held), attribute ID1803 (the number of color profiles held now), About other attributes ID other than attribute ID1804 (color profile table) and model ID180 (color profile tabular format), it is the same as that of what is used in an attribute table of Supervisor of drawing 7. However, a range which a range and influence of a setup attain to is limited to a thing under management of a color profile manager to whom an attribute table belongs. "A downloading method of data currently supported" of attribute ID601, And "an uploading method of data currently supported" of attribute ID602 is the same as that of what is during a print job manager's explanation and a scanning job manager's explanation, and is explained respectively, A downloading method and an uploading method of font data which are supported are shown.

[0081](Color profile table) Drawing 39 expresses a color profile which a color profile manager holds. This table is held DISK315 as an attribute value of attribute ID1804 (color profile table) of an attribute table shown by drawing 37. A color profile table means what kind of color profile a color profile manager has managed now. In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records. Each record comprises the filename 2703 containing color profile ID2701, the color profile format 2702, and color profile data. Color profile ID2701 is an identifier to a color profile which a color profile manager assigned, when color profile data downloads. Color profile data is correction data for amending data sent from Scanner Engine102, and data sent to Ink Jet Printer Engine105, and performing color matching.

[0082](Attribute table of a printer controller) Drawing 40 expresses data (attribute table) which each printer controllers 510, 511, and 512 hold to DISK315. This table expresses performance and a function of a print engine and a finisher which a printer controller is controlling, and rewriting of a value about performance and a function is impossible. In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records. The contents of each record are the same as that of a thing of Supervisor shown by drawing 7, and comprise attribute ID2801, model ID2802, and the value 2803. Drawing 41 shows the attribute ID of an attribute table of drawing 40, and a value of the model ID. Attribute ID12 (controller type), attribute ID13 (controller ID), Attribute ID5001 (can it color-print or not?), attribute ID5002 (kind of finishing currently supported), Attribute ID5003 (the highest resolution which can be set up), attribute ID5004 (the minimum resolution which can be set up), About the attributes ID and the models ID other than attribute ID5005 (job queue table) and model ID500 (job queue tabular format), it is the same as that of what is used in an attribute table of Supervisor of drawing 7. However, a range which a range and influence of a setup attain to is limited to a thing under management of a printer controller with which an attribute table belongs.

[0083]The service ID is an attribute table (printer controller 510) of 21, service ID22 and an attribute table of the printer controllers 511 and 512 corresponding to each of 23 are memorized by DISK315, and attribute tables of a printer controller shown in drawing 40 differ, respectively.

CONTINUE

Due to excessive amounts of data, whole translation was NOT completed.
For subsequent translation(s), please click on the above "CONTINUE" button.
When continued, the current translation will be overwritten with the new translation.

[Translation done.]

Continued translation.

[0084](Job queue table) Drawing 42 expresses a job queue table which all the printer controllers hold. This table is held DISK315 as an attribute value of attribute ID5005 (job queue table) of an attribute table shown by drawing 40. It is meant in what kind of state a job queue table has a job which a printer controller manages and performs. In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records. Each record comprises the filename 2903 by which substance of job ID2901, the job status 2902, and a job is held. Job ID2901 is an identifier to a job which a printer controller assigned, when a job is supplied to a printer controller. Drawing 43 is a figure showing the job status 2902 (state of a job) of drawing 42. 1 shows that it is in an execution suspension (compare) state according [two / job / an execution waiting state and 4] to an interruption job under execution and in 3 with an engine during end processing of a job. The filename 2903 of a job is a name of a file by which substance of a job is held. Substance of a job is constituted by having two or more groups of the attribute ID, attribute value size, and an attribute value, as shown in drawing 52.
[0085](Job control statement text table) Drawing 85 expresses a job control statement text table which all the printer controllers hold. The job ID (1563) in which a printer controller published this table to each print job manager and a copy job manager, The bottles ID (1566) of a file pointer (1564) in which a position in a supplied job file is shown, ** copy number of sheets (1565), and a finisher are the contents.
[0086](Attribute table of a scanning controller) Drawing 44 expresses data (attribute table) which the scanner controller 420 holds to DISK 315. This table expresses scanner engine performance and a function which a scanner controller is controlling.
Rewriting of a value about performance and a function is impossible.
In a table, each line expresses one information unit (record), and data is constituted as a set of two or more records.The contents of each record are the same as that of a thing of Supervisor shown by drawing 7, and comprise attribute ID3001, model ID3002, and the value 3003. Drawing 45 shows the attribute ID of an attribute table of drawing 44, and a value of the model ID. Attribute ID12 (controller type), attribute ID13 (controller ID), Attribute ID6001 (is a color scan possible or not?), attribute ID6002 (the maximum manuscript size), About other attributes ID other than attribute ID6003 (the highest resolution which can be set up), and attribute ID6004 (the minimum resolution which can be set up), it is the same as that of what is used in an attribute table of Supervisor of drawing 7.
However, a range which a range and influence of a setup attain to is limited to a thing under management of a printer controller with which an attribute table belongs.
[0087](Structure of a command packet), [drawing 46] As opposed to the user interface manager

405, the TCP/IP-UDP/IP processing module 406, the IEEE1284.4 processing module 407, and the SBP-2 processing module 408 to INTAPURINTA 409. Structure of a command packet outputted is expressed. This command packet From the interpreter 409 to the user interface manager 405. Structure of a reply packet outputted to the TCP/IP-UDP/IP processing module 406, the IEEE1284.4 processing module 407, and the SBP-2 processing module 408 and an event packet is also expressed. [a packet] . [a head of a packet] In order that the packet header 3101 to express, the packet version 3102 showing a structure version of a packet, the flag 3103 showing character of a packet, the operation code 3104 showing what kind of operation is performed, and Client (PC) may recognize a reply packet. The use ID 3107 used for attestation of the block number 3105 used being alike, the length table **** parameter length 3106 of the parameter 3110, and a user, the password 3108, the status code 3109 which is used only for a reply packet and expresses a general state of a reply. And it comprises the parameter 3110 with a format for which it opted every operation code 3105. The service ID for access, the attribute ID for access, etc. are included in the parameter 3110. The thing 3111 which expresses with the flag 3104 whether a packet is a command packet event packet or it is a reply packet, and data which should be transmitted to the parameter 3110 have not entered, Next, there is the continuation flag 3112 which shows that data of a continuation is contained also in a packet transmitted. The length of the parameter 3110 is restricted to a number (64 K bytes) by which the parameter length 3106 is denoted. An IP address and a port number are specified, it connects with the multifunctional peripheral equipment 201, and, specifically, ClientPC202 connected to Ethernet208 and 203 output IP packet data to Network Interface305 (107). ClientPC204 connected to the IEEE 1394 interface 206 specifies the nodes ID and LUN (logical unit number). It connects with the multifunctional peripheral equipment 201, and SBP-2 packet data is outputted to IEEE 1394 Interface306 (108). ClientPC205 connected to the IEEE1284 interface 207 specifies a socket number, connects with the multifunctional peripheral equipment 201, and outputs IEEE1284.4 packet data to IEEE1284 Interface307 (109). Network Interface Driver402, IEEE 1284 Interface. . [a transport packet by which IP Header, 1284Header, and 1394 Header were added to a head of a command packet shown in drawing 46 by Driver403 and IEEE 1394 Interface Driver404, respectively] It outputs to the TCP/IP-UDP/IP processing module 406, the IEEE1284.4 processing module 407, and the SBP-2 processing module 408. The TCP/IP-UDP/IP processing module 406, The IEEE1284.4 processing module 407 and the SBP-2 processing module 408 extract a command packet which processes each inputted transport packet and is shown in drawing 46, and output it to Interpreter/Generater409.

[0088](Processing of a command packet) Drawing 47 expresses a process flow of a command packet in the multifunctional peripheral equipment 201. In Step 3201, a command packet which each transport processing modules 405, 406, 407, and 408 process data inputted from each interfaces 401, 402, 403, and 404, and is shown in drawing 46 is extracted. [an extracted command packet] It is inputted into the interpreter 409 from the TCP/IP-UDP/IP processing module 406, the IEEE1284.4 processing module 407, and the SBP-2 processing module 408 with information (connection form ID) and a subaddress of connection form that data was inputted. A subaddress in which Supervisor410 holds the interpreter 409 in Step 3202 : [by comparing the connection form ID and a subaddress which were inputted with reference to a service ID conversion table (drawing 9)] The service ID is obtained and it is confirmed whether data input is effective. As a result of a check, when data input is not effective, in Step 3204, a command packet is canceled and it ends. When data input is effective, in Step 3203, analysis of a command packet is conducted based on packet structure of drawing 46. Each item shown by drawing 46 is outputted as separate information independent, respectively as a result of analysis of a packet. In Step 3205, a security level corresponding to the service ID is acquired by referring to an access control table (drawing 14) based on the service ID.

[0089]In Step 3206, it is investigated whether a security level acquired in Step 3205 is 0 or 1. When a security level is 0 or 1, it inputs into the dispatcher 412, without checking a security level (security gate 411). When a security level is in Step 3206 except 0 or 1, in Step 3207, it is investigated whether a security level is 2. When a security level is 2, it is confirmed whether a use ID contained in a command packet in Step 3209 exists in a user authentication table (drawing 13). When a use ID is not contained, in Step 3210, error reply information is generated as a thing without authority, and by INTAPURINTA 409, a packet (error information is embedded in the status code 3109) is generated, and it transmits to command packet transmitting origin. When a use ID is contained, it inputs into the dispatcher 412. A transmitting agency address is extracted from a transport packet (Header). A use ID contained in a command packet in Step 3208 in Step 3207 when a security level was except two: Confirm whether a pair of a password exists in a user authentication table (drawing 13). A use ID: When a pair of a password is not contained, generate error reply information in 3211 as a thing without authority, by INTAPURINTA 409, generate a packet and transmit to command packet transmitting origin. When a use ID is contained, it inputs into the dispatcher 412. In Step 3212, [the dispatcher 412] The connection form ID that determined a manager of a distribution place based on the service ID, and data was inputted to a determined manager, and a subaddress, An address, an operation code, a block number, flag information, a use ID, a password, parameter length, and a parameter (a job execution priority which is the job management setting information mentioned above, job specification execution time, and job residual limit time are included) of a transmitting agency are distributed. And in Step 3213, each manager processes these information.

[0090](Access processing of an attribute table) Drawing 48 expresses processing when Client points to read-out and writing to an attribute table of each manager and each controller which the multifunctional peripheral equipment 201 holds. Reading and writing from an attribute table which each manager and each controller hold are performed by transmitting a suitable command packet to a subaddress which Supervisor has. The service ID for access and the attribute ID for access are included in a command packet for attribute table read-out as a parameter. An attribute value corresponding to the service ID for access, the attribute ID for access, and the attribute ID is included in a command packet for attribute table writing as a parameter. Packet data sent to the multifunctional peripheral equipment 201 from Client is processed by flow shown in drawing 47, and is distributed to Supervisor. In Step 3301, the operation code 3104 confirms whether be a code ("Get") for attribute value read-out. When an operation code is "Get", in Step 3301, the whole attribute table for access is acquired based on the service ID. An attribute table of Supervisor410 shown in drawing 7 when the

service ID is 0, An attribute table of a print job manager when the services ID are 1, 2, 3, 4, 5, 6, 7, 8, and 9, as shows drawing 18, When the service ID is 10, figure 22 An attribute table of the scanning job manager 419 as shows, An attribute table of a copy job manager when the services ID are 11, 12, 13, 14, 15, 16, 17, and 18, as shows drawing 24, An attribute table of the font manager 413 when the service ID is 101, as shows drawing 26, An attribute table of the form overlay manager 414 when the service ID is 102, as shows drawing 29, An attribute table of the log manager 415 when the service ID is 103, as shows drawing 32, An attribute table of the color profile manager 416 when the service ID is 104, as shows drawing 37, When an attribute table of a printer controller when the services ID are 21, 22, and 23, as shown in drawing 40, and the service ID are 24, an attribute table of the scanner controller 420 as shown in drawing 44 is acquired. The service ID of an attribute table of a print job manager who shows drawing 18 is an attribute table (print job manager 504) of four, An attribute table of the print job managers 501, 502, 503, 505, 506, 507, 508, and 509 corresponding to each of service ID1, and 2, 3, 5, 6, 7, 8 and 9 is memorized by DISK315. Similarly the service ID of an attribute table of a copy job manager who shows drawing 24 is an attribute table (copy job manager 608) of 18, An attribute table of service ID11 and the copy job managers 601, 602, 603, 604, 605, 606, and 607 corresponding to each of 12, 13, 14, 15, 16, and 17 is memorized by DISK315, respectively. Similarly, the service ID is an attribute table (printer controller 510) of 21, and, as for an attribute table of a printer controller shown in drawing 40, service ID22 and an attribute table of the printer controllers 511 and 512 corresponding to each of 23 are memorized by DISK315.
[0091]Then, in Step 3303, it is inspected whether a specified attribute value is acquirable. A value of "an attribute ID list which only an administrator can acquire" (attribute ID=105) of target attribute tables is acquired, and an inspection is conducted by investigating whether the attribute ID by which acquisition directions were carried out is included in this. Since a value is unacquirable if the attribute ID is included, an error reply packet is created in Step 3305, and this is transmitted and it ends. If the attribute ID is not included, the model ID and an attribute value which search an attribute table based on the attribute ID specified in Step 3304, and have the applicable attribute ID are acquired. In Step 3306, a reply packet is generated based on the model ID, and a reply packet of a step 3307 smell lever is transmitted, and it ends. When an operation code is not "Get" in Step 3301, in Step 3308, an operation code confirms whether be a code ("Set") for attribute value writing. When an operation code is "Set", in Step 3309, the whole attribute table for access is acquired based on the service ID. Then, in Step 3310, it is inspected whether a specified attribute value can be set up. A value of "an attribute ID list which only an administrator can set up" (attribute ID=104) of target attribute tables is acquired, and an inspection is conducted by investigating whether the attribute ID by which setting directions were carried out is included in this. Since a value is unacquirable if the attribute ID is included, an error reply packet is created in Step 3314, and this is transmitted and it ends. If the attribute ID is not included, the model ID and an attribute value which search an attribute table based on the attribute ID specified in Step 3311, and have the applicable attribute ID are acquired. In Step 3312, an attribute value specified based on the model ID is set up, and a reply packet which tells that a setup was successful is generated. A reply packet of a step 3313 smell lever is transmitted, and it ends. When

an operation code is not "Set" in Step 3308, the processing step 3315 of others according to an operation code which are mentioned later is performed, and it ends.

[0092](Service ID list inquiry processing) Drawing 49 expresses processing when Client asks a list (list of the service ID) of service which the multifunctional peripheral equipment 201 provides. A subaddress which an inquiry of a list of service transmits a suitable command packet to a subaddress which Supervisor has, and is shown by drawing 9: Carry out by reading and processing a service ID conversion table. A flow shown by drawing 49 details a step of Step 3315 (processing of other Operation Code) of a flow shown by drawing 48. In Step 3401, it is confirmed whether an operation code is a code for service list acquisition ("List Service"). When an operation code is not "List Service", the processing step 3402 of others according to an operation code which are mentioned later is performed, and it ends. In Step 3401, when an operation code is "List Service", in Step 3403, a list which extracted only a thing equivalent to the connection form ID used for an inquiry out of a subaddress:service ID conversion table is created. In Step 3404, each record of a list created at Step 3403 is inspected, A use ID which uses for an inquiry a record in which a use ID used for an inquiry does not correspond to the value when validity [the effective flag 804] and the effective user list 805 have a value when the invalid user list 806 has a value deletes a record applicable to the value from a list. In Step 3405, a reply packet including a list of services ID updated at Step 3404 is created, and a created reply packet transmits in Step 3406.

[0093](Subaddress inquiry processing) Drawing 50 expresses processing when Client specifies the service ID and asks information on a subaddress which is needed when using service which the multifunctional peripheral equipment 201 provides. A subaddress which an inquiry of a subaddress transmits a suitable command packet to a subaddress which Supervisor410 has, and is shown by drawing 9: Carry out by reading and processing a service ID conversion table. The service ID is included in an inquiry command packet of a subaddress as a parameter. A flow shown by drawing 50 details a step of Step 3402 (processing of other Operation Code) of a flow shown by drawing 49. In Step 3501, it is confirmed whether an operation code is a code ("Reserve") for subaddress acquisition. When an operation code is not "Reserve", the processing step 3503 of others according to an operation code which are mentioned later is performed, and it ends. From the service ID specified in the connection form ID used for an inquiry in Step 3502, and a parameter when an operation code was "Reserve" in Step 3501 to a subaddress: Search a record which scans a service ID conversion table and corresponds. . [record / which was searched in Step 3504] [an effective flag] [be / it / effective] When an invalid user list has [whether when an effective user list has a value a use ID used for an inquiry is during a list, and] a value, it checks for no use ID used for an inquiry during a list. When either is not O.K. as a result of a check, either, in Step 3506, an error reply packet is created as a thing without the right to access (right of use) to specified service, and a packet of a step 3507 smell lever is transmitted and it ends. As a result of a check in Step 3504, in O.K., a reply packet which contains a subaddress in Step 3505 is created, and any check transmits a packet of a step 3507 smell lever, and is ended.

[0094](Inquiry processing of the service ID of task type specification) Drawing 51 expresses a process flow at the time of asking the optimal service ID, in order to process service of a kind specified by specifying a kind of service among services which the multifunctional peripheral equipment 201 provides. [an inquiry of the service ID which specified a kind of service] Service ID shown by a subaddress:service ID conversion table which transmits a suitable command packet to a subaddress which Supervisor has, and is shown by drawing 9, and drawing 11: Carry out by reading and processing a task type conversion table. Combination (condition information) for limiting a task type for specifying a kind of service and service as a parameter is included in a subaddress ID inquiry command packet. Condition information is expressed as a list of pairs of the attribute ID and a value. A flow shown by drawing 51 details a step of Step 3503 (processing of other Operation Code) of a flow shown by drawing 50. In Step 3601, it is confirmed whether an operation code is a code for subaddress acquisition ("Get Service"). When an operation code is not "Get Service", the processing step 3603 of others according to an operation code which are mentioned later is performed, and it ends. When an operation code is "Get Service" in Step 3601, The connection form ID and a subaddress which were used for an inquiry in Step 3602: Create a list of services ID which has the connection form ID used for an inquiry from a service ID conversion table. In Step 3604, a list of what has the task type specified out of the above-mentioned list is created with reference to a service ID:task type conversion table. In Step 3605, each record of a list created at Step 3604 is inspected, A use ID which uses for an inquiry a record in which a use ID used for an inquiry does not correspond to the value when validity [the effective flag 804] and the effective user list 805 have a value when the invalid user list 806 has a value deletes a record applicable to the value from a list. In Step 3609, it deletes except a value corresponding to conditions from a list as compared with condition information included in a parameter, and an attribute table of each manager corresponding to the service ID under list. In Step 3606, check the number of records of a list changed in Step 3605, and when the number of records is 0, In Step 3611, demanded service creates an error reply packet as what not existing, in Step 3613, transmits a reply packet and is completed. When the number of records is in Step 3606 except zero, in Step 3607, it is confirmed whether the number of records is 1. When the number of records is 1, a reply packet including the service ID obtained in Step 3612 is created, and in Step 3613, a reply packet is transmitted and it ends. When the number of records is in Step 3607 except one, the present loading state is asked to each manager corresponding to the service ID under list in Step 3608. Each manager returns the number of jobs under processing as a loading state, and chooses a manager who has the minimum load out of this. In Step 3610, a reply packet including the service ID corresponding to a selected manager is created, and in Step 3613, a reply packet is transmitted and it ends.

[0095](Structure of a job) Drawing 52 shows an internal structure of a file (job file) holding substance of a job which each manager manages. A filename of a job file is held by the filename 1502 of a job table (drawing 20). An internal structure of a file holding substance of a job which each controller manages also shows structure of this file. This filename is held by the filename 2903 of a job queue table (drawing 42). Substance of a job is expressed by having a group of attribute ID3701, the attribute value size 3702, and the attribute value 3703 in succession two or more. When a job

contains data, a filename of a file which holds size of a filename as a value and an attribute value which express data as the attribute ID as shown in 3707, 3708, and 3709, and holds document data as an attribute value is held. In an attribute, a transmission method of data, a format of data (PDL etc. which are used), Information, including an address etc. which transmit an event notification when a kind and an event of an event occur, And information depending on a kind of job : In the case of a print job, copy number of copies, Specification of a font which is finishing-processing-specified and is used, specification of a form overlay to be used, a job execution priority that is job management setting information, job specification execution time, job residual limit time, etc. are contained.
[0096](Job script processing by each manager) Drawing 53 and drawing 54 express a process flow of a job script in each manager. A job script is constituted by series of a command packet shown by drawing 46.
It is specified as what starts with a "Job Start" operation code and finishes it as a "Job End" operation code.
A subaddress each packet which constitutes a job script is indicated to be by drawing 9: It is supplied to a subaddress shown by a service ID conversion table, and each manager is supplied widely by command packet process flow shown by drawing 47. A process flow shown by drawing 53 is a process flow in a case of creating a job file and a data file which process a command packet distributed to each manager, and are shown by drawing 52. Operation distributed to each manager confirms whether be the operation code by which an operation code is supported in Step 3801. A check is performed by comparing with a value of an "operation currently supported" attribute (attribute ID101) which each manager holds in an attribute table. When it is the operation besides a support as a result of a check, an error reply packet is created in Step 3816, and this is transmitted and it ends. If an operation code of Step 3801 is supported, it will confirm whether present job be under injection in Step 3802. A check performs by inspecting whether a flag (flag set up at Step 3806) is truth during a job injection memorized to a predetermined field of RAM302. As for a flag, an imitation is set up by default during this job injection. And a flag is an imitation during a job injection as a result of an inspection, and if it is not [job / be / it] under injection, in Step 3804, it will be confirmed whether an operation code is "Job Start". When there is no operation code at "Job Start" as a result of a check, an error reply packet is created in Step 3816, and this is transmitted and it ends.
[0097]a parameter (a job execution priority.) which the job ID is assigned in Step 3805 and is an option of Job Start operation if it is O.K. as a result of a check of Step 3804 Based on execution start time and residual limit time, an item is added to a job table (drawing 20), and a job file is newly created to DISK315. This specification execution time and residual limit time are specified at relative time or absolute time, as mentioned above, but. When specification execution time and residual limit time are specified at relative time, CPU301 calculates time absolutely from relative time specified as the present time at that time in this step 3805, and it registers with a job table of drawing 20. After that, in Step 3806, a flag is set up truly during a job injection and processing of a command packet is ended. In a check of Step 3802, when a flag is truth during a job injection, in Step 3803, it is confirmed whether an operation code is "Send". Operation code "Send" directs to a manager that

data which constitutes a job is contained in a parameter. When an operation code is "Send" in Step 3803, A continuation flag inputted with a command in Step 3807 is checked, when a continuation flag is truth, a field is newly added to the data file 3713 which has already existed in Step 3808, and in 3809, a parameter is written in the field and it ends. When a continuation flag is an imitation in Step 3807, in Step 3810, the data file 3713 is newly created to DISK315, and a parameter is written in a data file in Step 3811. Then, in Step 3812, a field is added to a job file, and the size 3708 of attribute ID3707 which expresses data in Step 3813, and a filename, and the newly created filename 3709 of a file are written in this field, and it ends. When an operation code is not "Send" in Step 3803, in Step 3814, it is confirmed whether an operation code is "JobEnd". When an operation code is "Job End", in Step 3815, end processing of generation of a job is performed and it ends. In end processing of generation of a job, it changes including processing of a closing of a job file, etc. with manager's kinds (a print job, a scanning job, a copy job, a font, a form overlay, a log, a color profile).

[0098]In Step 3814, when an operation code is not "Job End", in Step 3817, it is confirmed whether an operation code is "Set Job". Operation code "Set Job" directs to a manager that an attribute (the attribute ID and an attribute value) which constitutes a job is included in a parameter. When an operation code of a job is "Set Job", it writes in a field which added a field to a job file in Step 3818, and added the attribute ID, attribute value size, and an attribute value in Step 3819, and ends.

[0099]In Step 3817, when an operation code is not "Set Job", in Step 3820, it is confirmed whether an operation code is "Send Request". In "Send Request", an operation code directs transmission of data to a manager, and a transmission method of data is included as a parameter. When an operation code is "Send Request", processing for which it depended on each manager in Step 3821 is performed, and it ends. Processing in Step 3821 in the case of a scanning job manager, It transmits by a method which had data obtained by pointing to a scan of a manuscript according to an attribute value currently held at a job file for a scanner controller specified, In the case of a font manager, transmitting font data specified as an attribute value in a job file with a transmission method specified immediately etc. is included. When an operation code is not "Send Request" in Step 3820, in Step 3822, other operation codes are processed and it ends.

[0100](Job processing (1) in a print job manager: Registration to a job-scheduling table from reception of job data), [drawing 55] Processing of a portion to registration of a job from job reception to a job-scheduling table or the order job queue table of execution time is expressed among process flows of a job in 509 and 513 from the print job manager 501 who showed by drawing 5. A process flow of a job script which a print job manager shows to drawing 54 is performed, and after a new item is added to a job table at Step 3805 by inputting a "Job Start" operation code, CPU301 performs a process flow shown by drawing 55. That an attribute which shows a kind of PDL (Page Description Language) currently used for expression of data (document data) in Step 3901 is added into a job file Waiting, It waits until PDL Rasterizer (PDL Rasterizer417 or PDL Rasterizer418) which will be used if a kind of PDL is decided becomes usable. If PDL Rasterizer becomes usable at Step 3901, it will wait to add an attribute which shows a data receiving method in Step 3902 into a job file. If an attribute which shows a data receiving method is added into a job file, a receiving method of data will be checked in

Step 3903. When a receiving method of data is the method of accessing the exteriors (for example, Client PC on a network, etc.) of multifunctional peripheral equipment MFT201, a task which reads data specified in Step 3904 is generated, and it accesses outside. It waits to start reading of data in Step 3905 after that. In Step 3903, when data is contained in a job (in the case of data reception according [on drawing 54 and] to the "Send" operation code), it waits to start reception of data in Step 3906. When reading or reception of data is started in a step of Step 3905 or 3906, in Step 3907, an injection is started for data to PDL Rasterizer. Then, in Step 3908, image data is received from PDL Rasterizer. It is judged from a job table which whether it is the no by which a specification execution time option is specified as a job of a step 3909 smell lever shows to drawing 20. a job-scheduling table which is specified and showing a job in drawing 83 if it becomes -- not but, it is linked to the order job queue table of execution time shown in drawing 84 (Step 3911). Under the present circumstances, it is judged from a job table shown in drawing 20 whether specification of residual limit time is carried out simultaneously, it is specified -- if it becomes, comparison of specification execution time and residual limit time is performed, and the latter is larger than the former (time -- back) -- etc. -- when a contradictory setup has accomplished, the job is thrown away, without registering with a table. When a specification execution time option is not specified in Step 3909, a job is linked to the tail end of an execution priority where a job-scheduling table was specified at Step 3910. When an execution priority is not specified at the time of a job injection, 3 (the minimum priority) of a default priority is given.

[0101](A print job manager's job processing (2): Registration to a job-scheduling table from the order job queue table of execution time) A task in which a print job manager supervises the order job queue table of execution time periodically (henceforth) an execution time surveillance task -- calling -- it is running and processing which detects having reached at execution time when a job in front [this] was specified, and is registered into a job-scheduling table is made to perform Processing of an execution time surveillance task is shown in drawing 86. It is confirmed whether search was completed about all the jobs linked to the order job queue table of effective time in Step 3920. Processing is an end if it is the end of a check. If search is completed about no still linked jobs, the next job of a link is followed in Step 3921, and the execution time is acquired. A job once searched with updating a pointer which points out a job of a retrieval object whenever processing of Step 3921 starts search from a head job of a table and it turns around a loop overlaps, and does not search. Specification execution time of a job acquired in Step 3921 is compared with the present time. When specification execution time of a job has passed over the present time, since the job is in the state where it should perform, it is deleted from a link at Step 3923, and is succeedingly added to a job-scheduling table with the priority 1 at Step 3924. This processing will continue until comparison of execution time and the present time is performed about all the jobs of the order job queue table of execution time. If comparison about all the jobs is completed, an execution time surveillance task will be ended, but this task is started a sufficiently short cycle, although progress of time is supervised.

[0102](A print job manager's job processing (3) : from a job-scheduling table up to an end of execution) A task in which a print job manager supervises a job-scheduling table periodically (henceforth) a job scheduler task -- calling -- it running and, This task is performing processing shown in drawing 87 which takes out a job which detects that a priority of that a new job was added to this table or a job was changed, and should be performed next from a job-scheduling table, and is supplied to a print controller. A job scheduler task is performing processing which a priority is the highest among jobs registered into a job-scheduling table (link), and takes out a job of a head of a link by processing of Step 3930 to the step 3932 of drawing 87. In [in Step 3930, specifically set a retrieval object priority of a print job manager's job-scheduling table to 1, and] Step 3931. It is confirmed whether the range of a retrieval object priority is within the limits (in this example, the range of a priority is 1 to 3) of a priority of a job-scheduling table. Since it means that a job of waiting for execution is not registered into front [this] if it is outside a range, it ends as it is. It is the processing which looks for a job linked to a head among jobs linked to a priority which is Step 3932. If it does not exist, a retrieval object priority is **************ed in Step 3933, and it returns and re-searches to Step 3931. If a job is found in Step 3932, this job will be decided as a job which should be supplied to a print controller. A job detected at Step 3934 is deleted from a table, and a print controller succeedingly thrown in at Step 3935 is chosen. When this determination is beforehand decided by attribute ID2001 (Controller ID list which performs Job) of the print job managers 501-507 and an attribute table of 509 and 513, . [by specifying C by attribute ID2002 (is Controller auto select possible or not?) of the print job manager 508, and reading the contents (the attribute I and an attribute value of a job file which are shown in drawing 52) of the print job file] A printer controller which is needed for a print may be determined dynamically. In this case, for example, if color printing is specified with the attribute ID and an attribute value in a job file, will determine Ink Jet Contorller512 (CID23), and. LBP Controller510 (CID21) will be determined if use of Finisher is specified with the attribute ID and an attribute value. A job taken out at Step 3934 to a print controller (single or plurality) selected in Step 3936 is supplied. After an injection adds as a record the job ID received from the job ID, each print controller ID, and each print controller to a job request table of a print job manager in Step 3937. Then, a print job manager will delete a record corresponding from a job request table in Step 3939, if an end of a job is notified to waiting and an end of a job by each printer controller from a printer controller at Step 3938. When it confirms whether all the jobs requested from each printer controller in 3940 were completed and a job still remains within a printer controller, it returns to Step 3938 and waits for an end of a job with a printer controller. In Step 3940, when a job in all the requested printer controllers is completed, in Step 3941, a record of a job is deleted from a job table (drawing 20). At this time, event transmitting processing shown by drawing 61 is performed. In Step 3942, inside of a job file is inspected and it is confirmed whether there is any attribute value which directs event transmission about an end of a job in Step 3943. If there is nothing, processing will be ended, but in a certain case, a transmission method and a transmitting address are acquired by reading an attribute value in Step 3944. A transmitting packet of an event is created in Step 3945, and an event is transmitted to a transmission method and a transmitting address which were specified. In Step 3946, end processing of deletion of a job file data file, etc. is performed, and processing of a job is ended.

[0103](A print job manager's job processing (4): Deletion of a job with residual limit time specification) Residual limit time may be specified as one job table of an attribute table of a print job manager as a parameter at the time of a job injection.This is an attribute specified when requiring time after being supplied before [execution of a job] and you do not want to hold an important job in apparatus by a non-run state on secrecy. That is, what a print manager checks periodically a job as which this residual limit time was specified, and is in a non-run state exceeding limit time deletes. Although an unperformed job is all over a job-scheduling table or it exists in the order job queue of execution time, or a job queue of each print controller, In this example, a job which performed comparison with the present time only about that to which residual limit time is set by a job all over a job-scheduling table registered, and is beyond a limit performs processing deleted out of front [this]. A print job manager has started a task (residual surveillance task) periodically because of realization of such a function. Drawing 88 shows a process flow of this residual surveillance task. Comparison of the time and present time is performed about that to which Step 3950 begins search from a job with a high execution priority of a job-scheduling table, and residual limit time is set among each job (Step 3951). In this comparison, discovery of a thing exceeding residual limit time will delete that job also from a job table of a print job manager corresponding in Step 3953 which is deleted from a table and continues in Step 3952. This processing is performed to all the jobs of a job-scheduling table, and a residual surveillance task is ended itself. A print job manager is setting to a system so that a residual surveillance task may be started after definite-period-of-time progress again.

[0104](An interruption print job manager's job processing) Especially a thing that can process an interruption job among print job managers is called an interruption print job manager. 513 of drawing 5 interrupts and it is a print job manager. That this print job manager 513 is service ID=19, and this is task type =11 from drawing 11 and also task type =11 interrupt from drawing 12, and it turns out that it is a print. If an interruption print job manager also performs a process flow of a job script which a print job manager shows to drawing 54 like other print job managers and an injection job is added to a job table, processing of drawing 89 will be performed. ****** is confirmed as processing of an interruption job has already been first performed in Step 3960. An interruption print job manager serves as specification which does not allow a multiplex injection of an interruption job. Therefore, if it is [interruption] already under processing, a return value of a command packet is filled in as refusing a receipt of a job in Step 3962, and it will return to a publishing agency and will end. If it is not [interruption / be / it] under processing, in Step 3961, a flag is set during interruption processing, and a multiplex injection is blocked. Step 3963 to the step 3970 performs job processing of the usual print job manager shown at Step 3901 to the step 3908 of drawing 55, and completely same processing, and receives image data. At Step 3971, processing of a job is continued as it is, without registering with a job-scheduling table or the order job table of execution time. It sinks below a job for a print controller which determined a print controller used at Step 3971, and was chosen at Step 3972, and supplies with directions. When the usual print job manager supplies a job to a print controller, it interrupts and does not give directions. This point can call it a difference in a large interruption print job manager and the usual print job manager. Step 3973 to the future step 3982 is the same as Step

3937 to the step 3946 of job processing of the usual print job manager shown in drawing 87. It interrupts in Step 3983 of the last which ended a job, and a flag is cleared and it ends.

[0105](Job processing of a print controller) A job processing flow of a print controller is shown in drawing 90. A print controller is thrown in in a job from a print job manager or a copy manager, and starts the processing. A print controller takes out one job which is running a task which supervises change of a job queue when processing, and this task links to a job queue, performs a printing request to a print engine, and processes a job. A request of a job to a print engine is given in a unit (usually page unit) which can process a print engine, and it is performed, performing handshaking between a print engine and a print controller. A task which finished processing all the jobs linked to a job queue waits for an injection of a new job, and stands by. Now, at the time of an injection of a job to a print controller, execution mode specification is given as a parameter. Execution mode is interrupted with order-of-arrival execution mode, and there is execution mode. Specifying interruption execution mode and supplying a job interrupts, it interrupts with the print job manager 513, and is only the copy manager 609. In Step 6001 of drawing 90, a check in execution mode which is a parameter of a job injection is performed. When interruption execution is not specified, it is added to a job queue of a print controller at Step 6010. When interruption execution is specified, an interruption job besides the present is investigated in under processing at Step 6002. This is understood with checking a flag during interruption job execution of a print controller. At Step 6003, it blocks so that a flag may be set during this interruption job execution and other interruption jobs may not be processed to multiplex. A task which is carrying out handshaking to a print engine is made to suspend at Step 6004 at the same time it evacuates a context (refer to drawing 85) about a job under execution to an area for context evacuation in DISK315 with the present print engine. It interrupts at Step 6005 after that, and a context is initialized for a job. In Step 6006, a job is divided and supplied to a print engine in a unit which a print engine processes. At the time of a supplied end of one unit, a print controller updates a context, and it repeats it until it sinks below this and a job is completed (Step 6007). If execution of an interruption job is completed, in Step 6008, resume of the task which returned and was making a context of an evacuated job which was interrupted suspend from an evacuation area will be carried out, and also it will interrupt at Step 6009, and a flag will be cleared during job execution. A resumed task will resume after this a job which existed in the middle of execution. A print engine and a task which is carrying out handshaking are not conscious of execution of an interruption job having interrupted in the meantime. If an interruption job besides the present is under execution in Step 6002, an error reply packet is created at Step 6011, and to packet issue origin, this packet will be transmitted and it will end.

[0106](Job processing in a scanning job manager) Drawing 56 expresses a process flow of a job in the scanning job manager 419. A scanning job manager starts processing of a job, after an injection of a job is completed. For this reason, a scanning job manager starts job processing of drawing 56 as end processing of a job of Step 3815 in drawing 54. In Step 4001, a job is supplied to the scanner controller 420 and the job ID is received. The job ID received from the job ID, the scanner controller ID, and the scanner controller 420 to a job request table (drawing 21) in Step 4002 is recorded. If an end of a job from the scanner controller 420 is notified to waiting and an end of a job from the scanner controller 420, [in Step 4003] Scanned image data is received from the scanner controller 420, and a record of a job is deleted from a job table (drawing 20) in Step 4004. At this time, event transmitting processing shown by drawing 61 is performed. In Step 4005, what is pointing to a transmission method of data out of an attribute in a job is searched. When data transmission methods are the directions which transmit data as a script in Step 4006, in Step 4008, data is transmitted as a script with other attribute information. As for data transmission processing from apparatus, details are shown in drawing 62. When it is the directions which transmit data as reference in Step 4006, in Step 4007, data is saved in an inside of apparatus, and reference information over this is transmitted as a script with other attribute information. Inside of a job file is inspected in Step 4010, and it is confirmed whether there is any attribute value which directs event transmission about an end of a job in Step 4011. If there is nothing, processing will be ended, but in a certain case, a transmission method and a transmitting address are acquired by reading an attribute value in Step 4012. A transmitting packet of an event is created in Step 4013, and an event is transmitted to a transmission method and a transmitting address which were specified. In Step 4014, end processing of deletion of a job file data file, etc. is performed, and processing of a job is ended.

[0107](Job processing (1) in a copy job manager: Registration to a job-scheduling table from a request of a scanning job) Drawing 57 expresses a process flow of a job in the copy job managers 601-609. A copy job manager starts processing of a job, after an injection of a job is completed. For this reason, a job manager starts job processing of drawing 57 as end processing of a job of Step 3815 in drawing 53. In Step 4101, a job is supplied to the scanner controller 420 and the job ID is received. The job ID received from the job ID, the scanner controller ID, and the scanner controller 420 to a job request table (drawing 21) in Step 4102 is recorded. If an end of a job from the scanner controller 420 is notified to waiting and an end of a job from the scanner controller 420, [in Step 4103] Scanned image data is received from the scanner controller 420, and a record of (drawing 21) to a job is deleted from a job request table in Step 4104. It is judged from a job table which whether it is the no by which a specification execution time option is specified as a job of a step 4105 smell lever shows to drawing 20. If specified, a job will not be in a job-scheduling table and it will be confirmed whether when [this] linked to the order job queue table of execution time (Step 4106), specification of residual limit time is also carried out simultaneously, it is specified -- if it becomes, comparison of specification execution time and residual limit time is performed, and the latter is larger than the former (time -- back) -- etc. -- when a contradictory setup has accomplished, the job is thrown away, without registering with a table. When a specification execution time option is not specified in Step 4105, a job is linked to the tail end of an execution priority where a job-scheduling table was specified at Step 4107. When an execution priority is not specified at the time of a job injection, 3 (the minimum priority) of a default priority is given.

[0108](A copy job manager's job processing (2): Registration to a job-scheduling table from the order job queue table of execution time) A copy job manager is running an execution time surveillance task which supervises the order job queue table of execution time periodically, Processing which detects having reached at execution time when a job in front [this] was specified, and is registered into a job-scheduling table is made to perform. Processing of an execution time surveillance task is shown. Since processing of a copy job manager's execution time surveillance task is the same as that of a process flow of a print job manager's execution time surveillance task shown in drawing 86, explanation is omitted.

[0109](A copy job manager's job processing (3) : from a job-scheduling table up to an end of execution) A copy job manager is running a job scheduler task which supervises a job-scheduling table periodically, This task is performing processing which takes out a job which detects that a priority of that a new job was added to this table or a job was changed, and should be performed next from a job-scheduling table, and is supplied to a print controller. Since processing is the same as that of a process flow of a job scheduler task of a print job manager who shows drawing 87, explanation is omitted.

[0110](A copy job manager's job processing (4): Deletion of a job with residual limit time specification) Residual limit time may be specified as one job table of a copy job manager's attribute table as a parameter at the time of a job injection. This is an attribute specified when requiring time after being supplied before [execution of a job] and you do not want to hold an important job in apparatus by a non-run state on secrecy. That is, what a copy job manager checks periodically a job as which this residual limit time was specified, and is in a non-run state exceeding limit time deletes. Although an unperformed job is all over a job-scheduling table or it exists in the order job queue of execution time, or a job queue of each print controller, In this example, a job which performed comparison with the present time only about that to which residual limit time is set by a job all over a job-scheduling table registered, and is beyond a limit performs processing deleted out of front [this]. A copy job manager has started a task (residual surveillance task) periodically because of realization of such a function. since processing of drawing 88 of a copy job manager's residual surveillance task is the same as that of a process flow of a residual surveillance task of a print job manager boiled and shown, explanation is omitted.

[0111](An interruption copy job manager's job processing) Especially a thing that can process an interruption job among copy job managers is called an interruption copy job manager. 609 of drawing 6 interrupts and it is a copy job manager. That this copy job manager 609 is service ID=20, and this is task type =13 from drawing 11 and also task type =13 interrupt from drawing 12, and it turns out that it is a copy. If an interruption copy job manager also performs a process flow of a job script which a copy job manager shows to drawing 54 like other copy job managers and an injection job is added to a job table, processing of drawing 91 will be performed. Step 4130 to the step 4132 of the figure is the same as Step 3960 to the step 3962 of job processing of an interruption print job manager of drawing 89. Step 4133 to the step 4136 of drawing 91 is the same as that of processing from Step 4101 of job processing of a copy job manager of drawing 57 to Step 4104. Step 4137 to the step 4149 is the same as Step 3971 to the step 3983 of job processing of an interruption print job manager of drawing 89.

[0112]Then, a print job manager will delete a record corresponding to a job ended from a job request table in Step 4109, if an end of a job is notified to waiting and an end of a job by each printer controller from a printer controller at Step 4108. When it confirms whether all the jobs requested from each printer controller in Step 4110 were completed and a job still remains within a printer controller, it returns to Step 4108 and waits for an end of a job with a printer controller. In Step 4110, when a job in all the requested printer controllers is completed, in Step 4111, a record of a job is deleted from a job table (drawing 20). At this time, event transmitting processing shown by drawing 61 is performed. In Step 4112, inside of a job file is inspected and it is confirmed whether there is any attribute value which directs event transmission about an end of a job in Step 4113. If there is nothing, processing will be ended, but in a certain case, a transmission method and a transmitting address are acquired by reading an attribute value in Step 4114. A transmitting packet of an event is created in Step 4115, and an event is transmitted to a transmission method and a transmitting address which were specified. In Step 4116, end processing of deletion of a job file data file, etc. is performed, and processing of a job is ended.

[0113](Job processing download) Drawing 58 expresses a thing about a download function of data among process flows of a job in the font manager 413, the form overlay manager 414, the log manager 415, and the color profile manager 416. A job to these managers is for performing download and upload of data which each manager manages. Management of reference, deletion, etc. of data which each manager manages is performed by accessing an attribute table which each manager has as a command packet is supplied to a subaddress which Supervisor410 manages and it is shown by drawing 48. Each manager starts processing of a job about data download, after an injection of a job is completed. For this reason, each manager starts job processing about data download of drawing 58 as end processing of a job of Step 3815 in drawing 53. In Step 4201, a job file (drawing 52) is scanned and it is confirmed whether an attribute about a receiving method of data exists. When an attribute about a receiving method of data does not exist as a result of a check, in Step 4208, this job file performs end processing as that to which a process flow which is related with upload and shown in drawing 59 has already been performed. A job file is deleted in end processing in Step 4208. When an attribute about a receiving method of data exists in Step 4201, a receiving method of data is checked in Step 4202. When a receiving method of data is included in a job (in the case of data reception according [on drawing 53 and] to the "Send" operation code), data which data is already ending with reception and was received in Step 4205 is saved as a file. When a receiving method of data is data in the apparatus exterior, an external source specified in Step 4203 is accessed, data is acquired and acquired data is saved as a file in Step 4205. a management table (a case where he is the font manager 413 -- a font table (drawing 28).) where each manager manages file information saved in Step 4205 in Step 4206 It registers by writing [in the case of the form overlay manager 414 / in the case of a form overlay table (drawing 31) and the log manager 415] a new record in a color profile table (drawing 39) in the case of a log table (drawing 34) and the color profile manager 416. In Step 4207, end processing of deletion of a job file data file, etc. is performed, and processing of a job is ended.

[0114](Job processing upload) Drawing 59 expresses a thing about an upload function of data among process flows of a job in the font manager 413, the form overlay manager 414, the log manager 415, and the color profile manager 416. Each manager starts processing of a job about data upload, when the "Send Request" operation code which constitutes a job script is supplied. For this reason, each manager starts job processing about data upload of drawing 59 as processing depending on each manager of Step 3821 in drawing 54. In Step 4301, what is pointing to a transmission method of data out of an attribute in a job file (drawing 52) is checked. When data transmission methods are the directions which transmit data as a script in Step 4301, (NO) transmits as a script data by which tbe data were carried out within a job with other attribute information in Step 4303. As for data transmission processing from apparatus, details are shown in drawing 62. When it is the directions which transmit data as reference in Step 4301, reference information over data specified within a job is transmitted as a script with other attribute information.

[0115](Job management (1) in each manager), [drawing 60] When Client performs operator guidance (deletion of a job) to a job which the print job managers 501-509, the scanning job manager 419, and the copy job manager 601-608 in the multifunctional peripheral equipment 201 manage. A process flow performed is expressed. Operation to a job which each manager manages is performed by transmitting a suitable command packet to a subaddress which Supervisor has. The service ID for specifying an object job manager and the object job ID are contained in a command packet for job management as a parameter. A command packet sent to the multifunctional peripheral equipment 201 from Client is processed by flow shown in drawing 47, and is distributed to Supervisor410. A process flow shown in drawing 60 details a step of 3603 (processing of other Operation Code) in a flow shown by drawing 51. In each of Step 4401, Step 4403, Step 4405, and Step 4407, . [Supervisor410] [be / for job deletion / ("Cancel Job") / an operation code] . [be / ("Promote Job") / it / for job priority change] processing (respectively -- job deletion of Step 4402.) which is for job execution time change ("Change Exec Time"), or confirms ******** [for job residual limit time change] ("Change ExpireTime"), and corresponds Job priority change processing of Step 4404, job execution time change processing of Step 4406, and residual limit time change processing of Step 4408 are performed. When operation codes are not above any, either (in the case [Step 4407] of No), the processing step 4409 of others according to an operation code which are mentioned later is performed, and it ends. All job managers are not preparing the above-mentioned processing. For example, the scanning job manager 419 has been empty processing in which job priority change processing, job execution time change processing, and residual limit time change processing do not have the contents.

[0116](Job management (2) in each manager: Job deletion) As for job deletion of Step 4402 in drawing 60, details are shown in drawing 92. According to the service ID specified in a parameter in Step 4421 of drawing 92, a parameter is sent to each manager corresponding to the service ID. Each manager searches inside of a job request table (drawing 21) where each manager manages the specified job ID in Step 4422. When there is the job ID specified in a job request table in Step 4423, the controller ID of a controller which is performing the job ID and a job of a controller corresponding to the job ID specified in Step 4424 is acquired. In Step 4425, the job ID of a controller is specified to a controller corresponding to the controller ID, and deletion of a job is directed. A reply packet of a purport that a record of a job of the job ID which deletes a record which corresponds out of a job request table in Step 4426, and corresponds from a job table (drawing 20) at Step 4433 continuously was deleted, and deletion of a job was successful at Step 4434 is created, and it transmits. When a job specified all over a job request table at Step 4423 is not able to be discovered, it will search from a job-scheduling table (drawing 83) at Step 4427. When a job of specification all over a job-scheduling table is found (Step 4428), in Step 4429, a specification job is deleted from a job-scheduling table, and it progresses to Step 4433. When a job is not able to discover a job specified all over a job-scheduling table at Step 4428, it will search from the order job queue table (drawing 84) of execution time at Step 4430. When a job of specification all over the order job queue table of execution time is found (Step 4431), in Step 4432, a specification job is deleted from the order job queue table of execution time, and it progresses to Step 4433. When a job specified at Step 4431 is not able to be discovered, as what a job in a controller has already ended, a specification job creates a reply packet of a purport that it did not exist, and transmits.

[0117](Job management (3) in each manager: Job priority change processing) As for job deletion of Step 4404 in drawing 60, details are shown in drawing 93. According to the service ID specified in a parameter in Step 4451 of drawing 93, a parameter is sent to each manager corresponding to the service ID. Each manager searches the specified job ID in Step 4452 out of a job-scheduling table (drawing 83) which each manager manages. When there is the job ID specified all over a job-scheduling table in Step 4453, it relinks to a priority after changing a link position of a specification job in a job-scheduling table in Step 4454. A reply packet which changed a job priority of the specified job ID in a job table (drawing 20) at Step 4457, and described a purport of an end of change at Step 4458 is created, and it transmits. When the job ID specified all over a job-scheduling table at Step 4453 is not able to be discovered, in Step 4455, the appointed job is searched out of the order job queue (drawing 84) of execution time. When the appointed job is found all over the order job queue table of execution time in Step 4456, it progresses to Step 4457. When a specification job is not able to be discovered in Step 4456, a specification job creates a reply packet which described that at Step 4458 since it was or execution had already been completed after already being supplied to a controller, and transmits.

[0118](Job management (4) in each manager: Execution time change processing) As for execution time change processing of Step 4406 in drawing 60, details are shown in drawing 94. According to the service ID specified in a parameter in Step 4471 of drawing 94, a parameter is sent to each manager corresponding to the service ID. Since it is while a job still exists in the order job queue (drawing 84) of execution time, a thing with an effective change of execution time searches with Step 4472 whether the appointed job exists in the order job queue of execution time. When a specification job is discovered at Step 4473, in Step 4474, execution time of a specification job in a job table is changed. Execution time can also be delayed [also advancing compared with the original appointed time, or]. In Step 4475, comparison of the present time and changed execution time is performed. It will reach at execution time and a ********** case will progress to Step 4476. In Step 4476, a specification job will be deleted from the order job queue of execution time, and it will register with a job-scheduling table (drawing 83) at Step 4477 instead. A priority in this case is linked as priority =1 of the highest priority. After that, at Step 4478, a reply packet of a purport that execution time change processing was successful is created, and it transmits. When not having reached at execution time yet in Step 4475, it does not generate but movement of a job progresses to Step 4478 as it is. When a specification job does not exist in the order job queue of execution time at Step 4473, at Step 4478, a specification job creates a reply packet which described that it was not able to discover, and transmits.

[0119](Job management (5) in each manager: Residual limit time change processing) As for execution time change processing of Step 4408 in drawing 60, details are shown in drawing 95. According to the service ID specified in a parameter in Step 4481 of drawing 95, a parameter is sent to each manager corresponding to the service ID. Change of residual limit time is effective, while a job still exists in the order job queue (drawing 84) of execution time or existing all over a job-scheduling table (drawing 83). Therefore, it is searched whether a job specified as either of both the tables in Step 4482, Step 4483, Step 4489, and Step 4490 exists. a case where a specification job is discovered by either of both the tables -- Step 4484 -- the job table middle finger -- a law -- residual limit time of a record of a job is changed. When discovered by neither of both tables, at Step 4488, a specification job creates a reply packet which described not having been discovered, and transmits. Residual limit time after being changed with the present time in Step 4485 is compared. If it has already passed over residual limit time, it will progress to Step 4486 and a specification job will be deleted. A job exists in either the order job queue table of execution time or a job-scheduling table. A reply packet of a purport that a record of a job of the job ID which corresponds from a job table at Step 4487 was deleted, and deletion of a job was successful at Step 4488 is created, and it transmits.

[0120](Event transmission) Drawing 61 expresses a process flow of event transmission in each manager. Each manager has an event setting table as shown in drawing 16 as a value of an attribute table which each one holds. As explanation of drawing 16 showed, when an event occurs in this table, connection form and an address which transmit an event are written. If a certain event occurs, each manager will recognize the event ID to an event. Then, in Step 4501, it is searched whether the event ID is registered with reference to an event setting table. In Step 4502, the event ID ends processing, if one is not registered. If at least the one event ID is registered in Step 4502, an address which transmits connection form used when transmitting an event at Step 4503, and an event will be acquired from the first record, and an event transmitting packet will be created in Step 4504. Into this packet, a parameter specified for every event ID and a parameter according to a format for every event ID registered into an event format table (drawing 17) which Supervisor410 holds are added. In Step 4505, it transmits to a transmitting address of connection form which acquired this event transmitting packet by 4503. And it judges whether from Step 4503 to the step 4505 was repeatedly processed about all records with the event ID at Step 4506, and if it is not processing, and it returns to Step 4503 and all the records are processed, it will end.

[0121](Data script transmission from multifunctional peripheral equipment) Drawing 62 expresses a process flow at the time of transmitting data from the multifunctional peripheral equipment 201 as a script which is continuation of a command packet. This process flow is used in the transmitting step 4008 of image data, the transmitting step 4303 of font data, etc. which are obtained as a result of a scanning job. In Step 4601, a packet including the attribute of data is generated and it transmits. If required, Step 4501 is repeated, and two or more attributes are transmitted. Data which had transmission specified is acquired in Step 4602. A command packet has the structure shown in drawing 46, and since size which can transmit as a parameter is restricted, in 4603, it inspects the length of data. When the length of data is over the length (64 K bytes) of a limit value as a result of an inspection, a command packet which added as a parameter data obtained by cutting data by the length of a limit value in Step 4606 in Step 4607 is generated, and it transmits. "Send" is set to this command packet as an operation code, and a continuation flag is set to it as truth. In Step 4608, the cut remaining data is acquired and data length of Step 4603 is inspected again. When data length is contained in the length of a limit value of a command packet as a result of an inspection of Step 4603, a command packet which added data as a parameter in Step 4604 is generated, and it transmits. "Send" is set to this command packet as an operation code, and a continuation flag is set to it as an imitation. In Step 4605, a command packet including the remaining attributes is generated and it transmits. If required, Step 4501 is repeated, and two or more attributes are transmitted and it ends.
[0122](Hard structure of Client PC) Drawing 63 is a figure showing ClientPC202 of this embodiment shown in drawing 2, and hardware constitutions (controller) in which 203, 204, and 205 were common. Inside a controller, the pointing devices 6004 of Memory(RAM)6002, CRT, etc., such as the display 6003, a keyboard, and a mouse, ROM6008, and DISK6009 are connected for CPU6001 via the bus 6010. Various programs and data which are shown in drawing 77 are memorized by DISK6009 (storage), such as a hard disk and a floppy disk.
It is read to Memory(RAM)6002 one by one if needed, and performs by CPU6001.
This DISK6002 could be removable to Client PC, or could be built in Client PC. A program shown in drawing 77 The network interface cable 208 (10 BASE-T), Composition which downloads from other Client PC and MFT201 via the IEEE 1394 interface cable 206 and the IEEE1284 interface cable 207, and is memorized by DISK6009 may be used. Hardware shown in drawing 63 constitutes general Client PC shown in drawing 78. When it displays when CPU6001 writes data in the display 6003, and CPU6001 reads data from the pointing device 6004, directions from a user are inputted.
[0123]The network interface connector 6005, the IEEE 1394 interface connector 6006, and the IEEE1284 interface connector 6007 are connected to the bus 6010.
The Ethernet (10 BASE-T) cable 208 shown in drawing 2, the IEEE 1394 cable 206, and the IEEE1284 cable 207 are connected.
When CPU6001 reads or writes in data from these interfaces, communication which uses each interface is performed.
[0124](Software configuration of Client PC) Drawing 64 expresses a part of Client202 using the multifunctional peripheral equipment 201, and software (control program) block diagram of 203, 204,

and 205. Software (control program) and data which Client PC uses are memorized by DISK6009 as shown in drawing 77. 501 is a user interface.
Each driver and a utility which are shown by this by 505 to 514 display information which the multifunctional peripheral equipment 201 has on the display 6003.
It is the stage which 502 is a data base (DISK6009) which holds information on the multifunctional peripheral equipment 201 in use (data) now, and Client connected to the multifunctional peripheral equipment 201, An apparatus information acquisition section of 503 acquires all the information which apparatus has, and holds in a database of 502. . 504 is used in each attribute table which the multifunctional peripheral equipment 201 holds. It is a regulation database (DISK6009) holding regulation information, including a meaning of each attribute, a data type for every attribute ID, a format of a parameter specified for every event ID, a task type meaning, a subaddress of Supervisor, etc. Each driver and a utility which are shown by 505 to 514 operate based on the regulation database 504 and the apparatus information database 502. 5050 generates a print job script with the directions from an application program which is a printer driver and operates on Client PC. 506 generates a scanning job script with the directions from an application program which is a scanner driver and operates on Client PC. 507 generates a copy job script with the directions from an application program which is a copy driver and operates on Client PC. 508 is a font management utility.
A job which performs download and upload of font data is generated, and a command which manages font data is generated.
509 is a form overlay management utility.
A job which performs download and upload of form overlay data is generated, and a command which manages a form overlay is generated.
510 is a log management utility.
A job which performs download and upload of log data is generated, and a command which manages a log is generated.
511 is a color profile management utility.
A job which performs download and upload of color profile data is generated, and a command which manages a color profile is generated.
512 is a job management utility.
A command which manages resumption of deletion and a stop / execution of a print job scan job copy job, etc. is generated.
513 is an equipment management utility.
A command which acquires a state of a printer controller scanner controller is generated.
514 is a fee collection management utility.
A command which billing data acquires is generated.

[0125]515 is a generator which generates a command packet based on a command generated from each driver shown by 505 to 514, and a utility. A generated command packet transmits to the

multifunctional peripheral equipment 201 using each transport processing module of 517, 519, and 521. The generator 515 interprets a reply packet and an event packet which were inputted from each transport processing module of 517, 519, and 521, and distributes them to a suitable driver or a utility. 517 is a processing module of SBP-2 which is a transport layer of IEEE 1394. 519 is a processing module of TCP/IP and UDP/IP. 521 is a processing module of IEEE1284.4 which is a transport layer of IEEE1284. 518 is an interface which processes a physical layer of IEEE 1394. 520 is a network interface. 522 is an interface which processes a physical layer of IEEE1284. Drawing 64 also expresses the user interface manager's 405 internal configuration in a software (control program) block diagram (drawing 4) of the multifunctional peripheral equipment 201 again. In the case of the user interface manager 405, The SBP-2 processing module 517 in drawing 64, TCP/IP and UDP/IP processing MOJU 519, the IEEE1284.4 processing module 521, the network interface 518, the network interface 520, The IEEE1284 interface 522 does not exist but the generator 515 is directly connected to the interpreter 409 in drawing 4.
An exchange of a command packet and a reply packet event packet is performed by this.
This direct connection does not exist in other Client202 and software (control program) composition of 203, 204, and 205.
[0126](Packet generation / transmitting processing) Drawing 65 expresses a process flow which the generator 515 generates a command packet shown by drawing 46 in response to a command or a job script from each driver shown by 505 to 514, and a utility, and transmits to the multifunctional peripheral equipment 201. In front of this process flow, it shall be beforehand connected to a suitable subaddress of multifunctional peripheral equipment of an addressee. Information with which subaddress it should connect is acquired by acquisition flow of apparatus information shown by drawing 66. A job script is defined by continuation of a command, and when a candidate for processing is a job script, it repeats from Step 4701 to the step 4706 until all the commands are processed. Issue of a job script as used in the following description means processing for every command from Step 4701 to Step 4707. The length of data transmitted with a command is inspected in Step 4701. When exceeding the length (64 K bytes) of a limit value of a parameter as a result of an inspection, in Step 4702, data is cut by the length of a limit value. Cut data generates a packet which contains an operation code corresponding to a command by making a continuation flag into truth in Step 4703, and a packet is transmitted in Step 4704. In Step 4701, when the length of data goes into the length of a limit value of a parameter, a packet which contains an operation code corresponding to a command by making a continuation flag into an imitation in Step 4705 is generated, and a packet is transmitted in Step 4706.
[0127](Information acquisition and software automatic composition in Client) Drawing 66 expresses a processing flow chart of the apparatus information acquisition section 503 of drawing 64. Client performs processing denoted by drawing 66 immediately after connection with the multifunctional peripheral equipment 201, acquires apparatus information, and holds the apparatus information acquisition section 503 to the apparatus information database 502 (DISK6009). This processing operates again, also when an event of configuration change is sent from apparatus. Client PC

connects with the multifunctional peripheral equipment 201 in Step 4801. An IP address and a port number are specified, it connects with the multifunctional peripheral equipment 201, and, specifically, ClientPC202 connected to Ethernet208 and 203 output IP packet data. ClientPC204 connected to the IEEE 1394 interface 206 specifies the nodes ID and LUN (logical unit number), connects with the multifunctional peripheral equipment 201, and outputs SBP-2 packet data. ClientPC205 connected to the IEEE1284 interface 207 specifies a socket number, connects with the multifunctional peripheral equipment 201, and outputs IEEE1284.4 packet data. A subaddress of a connection place uses a subaddress for Supervisor beforehand specified for every connection form used for connection. This default value is acquired from the regulation information database 504 (DISK6009). In Step 4802, a value of attribute ID100 (list of attributes ID) of an attribute table of Supervisor shown in drawing 7 is acquired. A command (Get) for attribute value acquisition is generated by making service ID0 of Supervisor, and the attribute ID into a parameter, and acquisition of an attribute value is performed by supplying a command to the generator 515. The generator 515 performs a process flow of drawing 65, and transmits a command packet for attribute value acquisition to the multifunctional peripheral equipment 201. In the multifunctional peripheral equipment 201, the above-mentioned command packet is received and an attribute value of the attribute ID specified by performing a process flow of drawing 47 and drawing 48 is replied to Client. In Client, a reply packet is analyzed with a generator and it sends to an apparatus information acquisition section. The attribute of attribute ID100 serves as a list of all the attributes ID in an attribute table of Supervisor. Repetition processing is carried out until it specifies the attribute ID in Step 4803, it acquires an attribute value and it acquires an attribute value of all the attributes ID at Step 4804, and the attribute value is saved with the attribute ID at the apparatus information database 502. All attribute tables (drawing 7) that Supervisor has in the above step are saved at the apparatus information database 502.

[0128]And a list of services ID (SID:TASK TYPE table shown in drawing 11) is acquired from an attribute table saved in Step 4805 at the apparatus information database 502 (DISK6009). Acquisition of the service ID can also be performed by generating a command (LISTSERVICE) for a list inquiry of the service ID, and supplying a command to the generator 515. At this time, a process flow shown by drawing 47, drawing 48, and drawing 49 is performed with the multifunctional peripheral equipment 201. Each manager's information is acquired for each [which was acquired from Step 4806 in Step 4805 in Step 4811] the service ID of every. Service ID1 (print job manager 501) which is the first service ID in the service ID excluding Supervisor (service ID0) from a list shown in drawing 11 at Step 4806 is acquired. A subaddress corresponding to the service ID is acquired at Step 4807. A command (Reserve) for subaddress acquisition is generated by making service ID into a parameter, and acquisition of a subaddress is performed by supplying a command to the generator 515. At this time, a process flow shown by drawing 47, drawing 48, drawing 49, and drawing 50 is performed with the multifunctional peripheral equipment 201. An acquired subaddress is saved at the apparatus information database 502. In Step 4808, a value (list of attributes ID) of attribute ID100 each manager's attribute table is acquired. A parameter is specified for the service ID and the attribute ID corresponding to each manager, a command (Get) for attribute value acquisition is generated, and acquisition of an attribute value is performed by supplying a command to the generator 515. Repetition processing is carried out until it specifies the attribute ID in Step 4809, it acquires an attribute value and it acquires an attribute value of all the attributes ID at Step 4810, and the attribute value is saved with the attribute ID at the apparatus information database 502. In Step 4811, it judges whether it is the service ID of the last of a list of services ID, and if it is not the last, in Step 4812, the service ID next to a list will be acquired from a list, and processing from Step 4807 will be repeated. In Step 4811. All the services ID of a list of services ID. the print job managers 501, 502, 503, 504, 505, 506, 507, 508, and 509, the scanning job manager 419, the copy managers 601, 602, 603, 604605, 606, 607, and 608, and the font manager 413. An attribute value of the form overlay manager 414, the log manager 415, the color profile manager 416, the printer controllers 510, 511, and 512, and the scanner controller 420 is saved with the attribute ID at the apparatus information database 502. and, Based on apparatus information (data) saved at the apparatus information database 502 (DISK6009), a user interface for a manager corresponding to each of each driver utilities (control program) 505-514 consists of Step 4813 automatically. Renewal of a user interface is included in processing of the automatic composition step 4813.

A display of a setting screen (user interface) is changed according to a driver utility.

[for example a user interface of a printer driver] [like drawing 81 in which a display screen displayed on the display 6003 is shown or drawing 82] A list of PDL currently supported based on an attribute value of each print job manager's (501-509) attribute ID1101, Based on attribute ID1102, color printing is possible or a display of the minimum resolution which can be set up is changed based on the highest resolution which can be set up, and attribute ID1105 based on a kind of Finishing currently supported based on an attribute value of attribute ID1103, and an attribute value of attribute ID1104. A display of a user interface for other drivers 506 and 507 and the utilities 508-514 is similarly changed according to an attribute. Icons, such as (a print manager, a copy job manager), etc. of each driver utility, are displayed identifiable. Finally, connection with Supervisor410 is ended in Step 4814, and processing is ended.

[0129](Issue of a job which specified a task type) Drawing 67 expresses processing when each application utility publishes a job script which specified a task type. Although carried out by transmitting a suitable job script to a subaddress obtained by issue of a job which specified the service ID searching the apparatus information database 502 by using service ID as a key, When only a task type is specified by the directions from a user, it is necessary to direct an introduction request of the service ID used to the multifunctional peripheral equipment 201. Drawing 67 shows processing in this case. Step 4901 It sets, a command for attribute value acquisition which specifies a task type and acquires the service ID is generated, and it carries out by supplying a command to the generator 515. In the multifunctional peripheral equipment 201, a process flow shown by drawing 47, drawing 48, drawing 49, drawing 50, and drawing 51 is performed. In Step 4902, a job script is transmitted to a subaddress obtained by searching the apparatus information database 502 based on the service ID obtained at Step 4901.

[0130](Structure of an event) Drawing 68 expresses structure of an event sent to Client from the multifunctional peripheral equipment 201. It is the structure of a portion of the parameter 3110 of a packet which an event packet has the same structure as a command packet shown in drawing 46, and is shown by drawing 68. 5001 is the event ID showing a kind of event. 5002 is parameter data specified for every event ID. A format of this data is held beforehand at the regulation information database 504. 5003 is parameter data the format is decided to be by apparatus for every event ID. A format of this data is held by performing a process flow of drawing 66 as an event format table (drawing 17) at the apparatus information database 502.

[0131](Processing of an event) Drawing 69 is a flow showing processing when each driver utility receives an event from the multifunctional peripheral equipment 201 in Client. Each driver utility registers an event [want / one] into the generator 515 beforehand by using the event ID as a key. If an event is sent to Client from the multifunctional peripheral equipment 201, the generator 515 will distribute an event to each registered driver utility. Drawing 69 expresses processing after each driver utility receives distribution of an event. In Step 5101, the regulation parameter data 5002 is analyzed based on format information acquired from the regulation information database 504. In Step 5102, the event format 1302 corresponding to the event ID is acquired from an event format table (drawing 17) currently held by using the event ID (5001) as a key at the apparatus information database 502. [in Step 5103] [based on an acquired event format] An event format (attribute ID676:paper size of an event format, a kind of ID756:paper, a kind of ID666:toner, a kind of ID698:ink, ID600: position of a cover) of apparatus dependence is analyzed. and a parameter and the event ID (the event ID200:paper nothing.) which were analyzed in Step 5104 ID399: Toner nothing, ID432:ink nothing, ID234: Process cover opening of multifunctional peripheral equipment and perform a display (for example, user interfaces, such as paper nothing, cover opening, and those without toner, are displayed on the display 6003) according to an event.

[0132](Processing of a log) Drawing 70 expresses a flow which the log management utility 510 acquires a log which the multifunctional peripheral equipment 201 holds, and is processed. Log data are acquired in Step 5201. A job script for log-data acquisition is generated, and acquisition of log data is performed by sending to the multifunctional peripheral equipment 201. In the multifunctional peripheral equipment 201, a process flow shown by drawing 47, drawing 53, and drawing 59 is performed, and log data are sent to Client. For example, LogFormatID1, and 98/1/3 which are shown in drawing 35, [log data "12345, 4 1.23, "O.K."" of 1:23] In the amount of toner which the number of papers which a use ID which published a job outputted by 12345 used by 4, 1.23 and an end state of a job show a print job manager of service ID1 O.K. In Step 5202, a log format table (drawing 36) is acquired from the apparatus information database 502. In Step 5203, log format ID2401 in a head of each record of log data is acquired, and the log format 2502 out of a log format table (drawing 36) is acquired from this value. For example, a format "1:701, 1:565, 1:765, 1:777" of LogFormatID1 shows a print job manager of service ID1 a use ID which published a job, the outputted number of papers, the used amount of toner, and an end state of a job. A format "11:701, 11:565, 11:765, 11:777" of LogFormatID2 shows a print job manager of service ID11 a use ID which published a job, the outputted number of papers, the used amount of toner, and an end state of a job. In Step 5204, analysis of the log data 2403 is conducted according to the log format format 2502, and analyzed data is processed with the log generating time 2402 (a log is sorted). And it judges whether log data of each record were analyzed in Step 5205, and processing will be ended, if it is not analyzing, and processing from Step 5203 is repeated and performed and all are analyzed.

[0133](Acquisition and processing of fee collection information) Drawing 71 expresses a flow in which the fee collection management utility 514 acquires and processes billing data which the multifunctional peripheral equipment 201 holds. Billing data is expressed as count data which each manager holds. Count data is held as a value of attribute ID401 in an attribute table (drawing 7, drawing 18, drawing 22, drawing 24, drawing 26, drawing 29, drawing 32, drawing 37, drawing 40, drawing 44), and the value serves as a list of integral values. A place which each integral value means is expressed as a value of the attribute (attribute ID402) of a count data format in each manager's attribute table.

A value serves as a list of attributes ID.

A list of count data and a list of count data formats correspond by turn under list. For example, the 3rd value of count data means information on the 3rd attribute ID of a count data format. Drawing 71 expresses a flow which acquires and processes count data which each manager holds. In Step 5301, the target manager's service ID is specified and a value (value of count data) of attribute ID401 is acquired. For example, a value (45, 78, 34, 13) of attribute ID401 shows print number of sheets of paper size denoted by a count data format of attribute ID402. A parameter is made into a parameter for the service ID and the attribute ID, a command for attribute value acquisition is generated, and acquisition of a value of this attribute ID401 is performed by supplying a command to the generator 515. In Step 5302, the same service ID is specified and a value (value of a count data format) of attribute ID402 is acquired. For example, in the attribute value 565 of attribute ID402, print number of sheets of the paper size A2 and the attribute value 537 mean print number of sheets of the paper size A3, the attribute value 545 means print number of sheets of the paper size A4, and the attribute value 523 means print number of sheets of the paper size A5. This may be acquired from the apparatus information database 502. In Step 5303, an acquired count data format is used and count data is analyzed. In Step 5304, analyzed count data is used, for example, for every service ID, every use ID, and every section, processing and generation of billing data are performed, and it ends.

[0134](Issue of a print job and a data download job), [drawing 72] Issue of a print job by the printer driver 505, FONTODETA download job script issue by the font management utility 508, Form overlay-data download JOBUSUKURIPUTO issue by the form overlay utility 509, Each processing of log-data download Jobs crypto issue by the log management utility 510 and KARAPUROFAIRUDETA download job script issue by the color profile management utility 511 is expressed. A process flow by each above software (control program) is common, Print object document data and data for a Dow load of the above-mentioned various kinds (font data, form data, color profile data, log data) are described to be "data" in drawing 72 and the following explanation. In the case of job issue, data printed or downloaded is specified with two or more attribute values set up into a job script. In Step 5401, the apparatus information database 502 is accessed and apparatus information is acquired. In

Step 5402, a specified attribute value confirms whether be within the limits of each attribute value set up in an attribute table with reference to an attribute table which each manager (for example, print manager which specifies printing if it is print object document data) whom data makes applicable to an injection has. When there is no at least one in within the limits as a result of a check, processing of job issue refusal is performed and displaying an error dialog on a user interface in Step 5405 etc. is completed.

[0135]When all the attributes are within the limits of each attribute value in an attribute table which a manager has as a result of a check of Step 5402, it is confirmed whether, in Step 5403, a prohibition setting attribute (from attribute ID801 to 805) exists all over a manager's attribute table. When a manager's attribute table has a prohibition setting attribute, it confirms whether be the combination of a preset value to which the attribute of a job specified for each [a manager has in an attribute table in Step 5404] prohibition setting attribute of every is forbidden. When there is combination of a preset value of an attribute forbidden as a result of a check, in Step 5405, job issue refusal is processed and it ends. When a manager does not have a prohibition setting attribute, and when it is not the combination of a preset value to which the attribute of a job is forbidden as a result of a check of Step 5404 as a result of a check of Step 5403, a data's existence place is checked in Step 5406. Specification of a data existence place is performed because ** and a user specify an address or application generates an address. In inside Client in which a data's existence place publishes a job, in Step 5407, the attribute value 601 (downloading method of data currently supported) of an attribute table of each person in charge manager in the apparatus information database 502 is checked.

[0136]In a check of Step 5407, when a method of including data in a job is supported, a job script in which data was included in a job in Step 5408 is generated, and it transmits to the multifunctional peripheral equipment 201 through packet generation processing shown in drawing 65. In [when a method of including data in a job is not supported in a check of Step 5407] Step 5409, A job script in which a reference pointer (URL:Unified Resource Locator) of data which exists in apparatus was included in a job is generated, and it transmits. In a check of Step 5406, when data exists in the Client exterior which publishes a job, in Step 5410, the attribute value 601 (downloading method of data currently supported) of an attribute table of each person in charge manager in apparatus information data is checked. In a check of Step 5410, when a way include a reference pointer and apparatus itself reads data in a job is supported, a job script in which a reference pointer to data was included in a job is generated, and it transmits. In a check of Step 5410, when a way include a reference pointer and apparatus itself reads data in a job is not supported, in Step 5412, data is once read into an inside of apparatus from the apparatus exterior. And in Step 5413, a job script which contains read data in a job is generated, and it transmits. [each manager of the multifunctional peripheral equipment 201 who received a job script] Processing shown in drawing 53, drawing 54, drawing 55 (print job manager) or drawing 53, drawing 54, and drawing 58 (a font manager, a form overlay manager, a log manager, a color profile manager) is performed, and a directed job is processed.

[0137](Issue of a scanning job and a data upload job), [drawing 73] Issue of a scanning job by the scanner driver 506, FONTODETA upload job script issue by the font management utility 508, Form overlay-data upload JOBUSUKURIPUTO issue by the form overlay utility 509, Each processing of log-data upload Jobs crypto issue by the log management utility 510 and KARAPUROFAIRUDETA upload job script issue by the color profile management utility 511 is expressed. A process flow by each above software (control program) is common, Document data and data for upload of the above-mentioned various kinds (font data, form data, color profile data, logo data) which were obtained as for the scanning result are described to be "data" in drawing 73 and the following explanation. In the case of job issue, a directions child indicating data is specified with two or more attribute values set up into a job script. In Step 5501, the apparatus information database 502 is accessed and apparatus information is acquired. Each manager who makes data applicable to upload in Step 5502. With reference to an attribute table which (for example, a font manager who manages font data if it is font data) has, a specified attribute value confirms whether be within the limits of each attribute value set up in an attribute table. When there is no at least one in within the limits as a result of a check, processing of job issue refusal is performed and displaying an error dialog on a user interface in Step 5505 etc. is completed. When all the attributes are within the limits of each attribute value in an attribute table which a manager has as a result of a check of Step 5502, it is confirmed whether, in Step 5503, a prohibition setting attribute (from attribute ID801 to 805) exists in a manager's attribute. When a manager's attribute table has a prohibition setting attribute, it confirms whether be the combination of a preset value to which the attribute of a job specified for each [a manager has in an attribute table in Step 5504] prohibition setting attribute of every is forbidden. When there is combination of a preset value of an attribute forbidden as a result of a check, in Step 5505, job issue refusal is processed and it ends.

[0138]When a manager does not have a prohibition setting attribute, and when it is not the combination of a preset value to which the attribute of a job was forbidden as a result of a check of Step 5504 as a result of a check of Step 5503, a transmitting address (preservation position) of data is checked in Step 5506. When a transmitting address of data is the apparatus exterior, in Step 5507, a transmitting address of data is checked again. In Client in which a transmitting address of data publishes a job, in Step 5508, the attribute value 602 (uploading method of data currently supported) of an attribute table of each person in charge manager in apparatus information data is checked. When data transmission is being supported as a result of a check of Step 5508, in Step 5509, a job script which directs data transmission is published and data reception is performed in Step 5510. When data transmission is not being supported as a result of a check of Step 5508, [(NO)] A job script which directs reference transmission which means detaining data and transmitting a reference pointer to data as a reply into apparatus in Step 5511 is published, and data is acquired in Step 5512 using information on a replied reference pointer. In other than Client in which a transmitting address of data publishes a job, in Step 5507, the attribute value 602 (uploading method of data currently supported) of an attribute table of each person in charge manager in apparatus information data is checked in Step 5513. When reference transmission is being supported as a result of a check of Step 5513, Acquisition directions of data are sent by publishing a job script which directs reference transmission which makes a transmitting address a parameter in Step 5514, and telling that data is transmitted from the multifunctional peripheral equipment 201 to a transmitting address of data in Step 5515. When reference transmission is not being supported as a result of a check of Step 5513, a job script which directs data transmission in Step 5516 is published, data is received in Step 5517, and this data is again transmitted to a transmitting address. When a transmitting address is an inside of apparatus as a result of a check of Step 5506, in Step 5518, the attribute value 602 (uploading method of data currently supported) of an attribute table of each person in charge manager in apparatus information data is checked. When reference transmission is being supported as a result of a check of Step 5518, light is emitted in a job script which directs reference transmission which makes an address in apparatus a parameter in Step 5519. When reference transmission is not being supported as a result of a check of Step 5518, a job which directs data transmission in Step 5520 is published, and after Client receives data in Step 5521, data is again returned to apparatus. [each manager of the multifunctional peripheral equipment 201 who received a job script] Processing shown in drawing 53, drawing 54, drawing 56 (scanning job manager) or drawing 53, drawing 54, and drawing 59 (a font manager, a form overlay manager, a log manager, a color profile manager) is performed, and a directed job is processed.

[0139](Issue of a copy job) Drawing 74 expresses processing which publishes a copy job. As directions of copy job issue, two or more attribute values set up into a job script are specified. In Step 5601, the apparatus information database 502 is accessed and apparatus information is acquired. In Step 5602, a specified attribute value confirms whether be within the limits of each attribute value set up in an attribute table with reference to an attribute table which a copy manager has. When there is no at least one in within the limits as a result of a check, processing of job issue refusal is performed and displaying an error dialog on a user interface in Step 5605 etc. is completed. When all the attributes are within the limits of each attribute value in an attribute table which a copy manager has as a result of a check of Step 5602, it is confirmed whether, in Step 5603, a prohibition setting attribute (from attribute ID801 to 805) exists all over a copy manager's attribute table. When a copy manager's attribute table has a prohibition setting attribute, it confirms whether be the combination of a preset value to which the attribute of a job specified for each [a manager has in an attribute table in Step 5604] prohibition setting attribute of every is forbidden. When there is combination of a preset value of an attribute forbidden as a result of a check, in Step 5605, job issue refusal is processed and it ends. When a manager does not have a prohibition setting attribute, and when it is not the combination of a preset value to which the attribute of a job is forbidden as a result of a check of Step 5604 as a result of a check of Step 5603, in Step 5606, a job script is generated and published and it ends. A copy job manager of the multifunctional peripheral equipment 201 who received a job script performs processing shown in drawing 53, drawing 54, and drawing 57, and manages a directed job.

[0140](Issue of a job management command) Drawing 75 expresses a process flow in case the job management utility 512 publishes a job management command. In Step 5701, a job table (drawing 20) which a manager holds by making the target manager's service ID into a parameter is acquired from MFT201. In Step 5702, the job ID which a job made applicable to management by a suitable method out of a job list included in an acquired job table has is chosen. As a selecting means, a list of jobs is displayed on the display 6003 via the user interface 501, and a method etc. which a user is made to choose are included. In Step 5703, a command for job management which makes a parameter the specified job ID is generated and published, and it ends. Supervisor of the multifunctional peripheral equipment 201 which received a job management command performs drawing 47, drawing 48, drawing 49, drawing 50, drawing 51, drawing 60, and processing shown in drawing 95 from drawing 92, and manages a directed job. A job management command of Step 5703 of drawing 75 specifically, respectively "CancelJob", When it is "Promote Job", "Change Exec Time", and "Change Expire Time", job management processing by a job manager corresponding to each of it performed Job deletion of drawing 92, job priority change processing of drawing 93, They are execution time change processing of drawing 94, and residual limit time change processing of drawing 95.

[0141](Issue of an interruption job) Drawing 96 shows a dialog which it displays on the display 6003 when the printer driver 505 publishes an interruption print job. When a user wedges himself and it points to a print when supplying a job, the dialog 5000 which interrupt and to which an option is made to set appears on a screen of client PC. In this dialog, it is set up whether you make what give priority to and interrupt as an option of an interruption print. Namely, the multifunctional peripheral equipment 201 has two or more print engines, and some selections are possible for it by difference between speed, resolution, a FINISSHIINGU function, etc.

[0142]For example, since liking to make waiting time to an end of a print into the shortest or the image quality cannot reach a compromise, even if it takes output time a little, liking of a user of image quality wanting to choose a good printer can be set up. Directions are issued so that a printer which is satisfied with the first priority a user's liking specified here may be chosen and a job under execution of them (multiple selection can be made) may be interrupted. One favorite option of a user is chosen from 5005 of drawing 96 in 5007. 5004 is interrupted and a candidate of a print engine who can perform a job is listed. A favorite thing can be chosen from a print engine in which 5004 was listed when a user chooses the print engine specification 5007 in a setup of 5003. In an example of drawing 96, since a user chose the end time priority 5005, all the printers listed will be used and he will do the distributed output of the job. If a setup is completed, it will push and sink below a print execution button of 5001, and a job will be published.

[0143]An end dialog of a job of drawing 97 is displayed on user PC ignited by a notice event of an end of an interruption job published from the multifunctional peripheral equipment 201. A used print engine name which is indicated to be interruption specification of a user who shows 5022 to 5023, and the outputted number of pages are displayed in a list on this dialog. In this example, since a user printed by interrupting by end time priority, he performed load sharing so that end time might become the shortest in three print engines which were possible for job processing at that time. A user will collect documents outputted according to this list.

[0144]Like drawing 98, if a user chooses an option of printer specification, 6004 is interrupted and information (attribute value of an interruption print job manager's attribute) about an interruption print job manager who can perform a job may be listed. A virtual (it is logical) printer name is attached to each interruption print job manager, and performance and functions of the printer are enumerated in a parenthesis. Here, an attribute value of an interruption print job manager's attribute is acquired at Step 5401 of drawing 72.

[0145]a user -- any of printer engine of drawing 97 -- ****, if either of the virtual printers of drawing 98 is chosen, A subaddress obtained by searching the apparatus database 502 is interrupted by using service ID of an interruption print manager corresponding to each as a key, and a job script of a print job is transmitted.

[0146]It may enable it to specify an interruption print on a screen like drawing 99. A list of a print manager contained in a list of services ID acquired at Step 4806 of drawing 66 and interruption print managers is expressed as a screen of drawing 99. deep -- coming out -- a manager name is attached about a print manager and an interruption print manager, and a list indication of the manager name is given. A user wedges himself out of a manager's list and chooses a print manager to perform an interruption print. Then, a subaddress obtained by searching the apparatus database 502 is interrupted by using service ID of the interruption print manager as a key, and a job script of a print job is transmitted.

[0147]On a screen, a printing method of a print manager or an interruption print manager, paper size, propriety of color printing, propriety of double-sided printing, the range of possible expansion/reduction, etc. are displayed based on an attribute value acquired at Step 4809, for example. The user can know whether seeing this attribute value, it is what kind of printing method, and it interrupts in what kind of paper size, and a print is made, an interruption print of color printing is made, or an interruption print of double-sided printing will be made.

[0148]Here, a printing method is called for by which print controller is used from an attribute value of attribute ID2001 (list of Controller ID which performs job) of a print job manager or an interruption print manager. Propriety of color printing is calculated from an attribute value of attribute ID1102 of a print job manager or an interruption print manager. Detailed information is able to be displayed based on an attribute value.

CONTINUE

Due to excessive amounts of data, whole translation was NOT completed.
For subsequent translation(s), please click on the above "CONTINUE" button.
When continued, the current translation will be overwritten with the new translation.

[Translation done.]

Continued translation.

[0149]Even if it applies this invention to the system which comprises a copy machine, a printer, a scanner, etc. which exist alone, it may be applied to the devices (for example, a copying machine, a facsimile machine, etc.) which consist of one apparatus. [the purpose of this invention] [the storage (drawing 76, drawing 77) which recorded the program code of the software (control program) which realizes the function of an embodiment mentioned above] As shown in drawing 2, a system is supplied, and it is attained when the device (CPU301 and CPU6001) of the system reads and executes the program code stored in the storage. The method of making floppy disk FD7000 memorize and supplying to the main part 7001 of PC, as shown in drawing 78 as a method of supplying the program shown in drawing 77 and data to Client PC is also common. In this case, the function of an embodiment which the program code itself read from the storage mentioned above will be realized, and the storage which memorized that program code will constitute this invention.
[0150]As a storage for supplying a program code, an optical disc, a magneto-optical disc, CD-ROM, CD-R, magnetic tape, the memory card of fixity, ROM, etc. can be used, for example besides a floppy disk or a hard disk. [by executing the program code which the computer read] The function of an embodiment mentioned above is not only realized, but, [based on directions of the program code] It cannot be overemphasized that it is contained also when the function of an embodiment which performed a part or all of processing that OS (operating system) etc. which are working on a computer are actual, and was mentioned above by the processing is realized. After the program code read from the storage was written in the memory with which the expansion unit connected to the expansion board inserted in the computer or the computer is equipped, It cannot be overemphasized that it is contained also when the function of an embodiment which performed a part or all of processing that CPU etc. with which the expansion board and expansion unit are equipped are actual, based on directions of the program code, and was mentioned above by the processing is realized.
[0151][the purpose of this invention] [the program code (drawing 76, program code of drawing 77) of the software (control program) which realizes the function of an embodiment mentioned above] [the memory measure (Memory302, Disk315, Memory6002, Disk6009) of a system as a program sending device shows to drawing 2] It supplies via LAN or a public network, and it is attained also when the device (CPU301 and CPU6001) of the system reads and executes the program code stored in the memory measure. A program sending device is mainly a server in the system of a client server method.
For example, a web server, a ftp server, etc. are mentioned.
In this case, the function of an embodiment which the program code itself which the program sending

device sent out mentioned above will be realized, and the sending device which sent out that program code will constitute this invention.

[0152]Drawing 79 is a sectional view showing the internal structure of a laser beam printer (it abbreviates to LBP hereafter) applicable to the laser beam printer engines 103 and 104 of drawing 1, and this LBP can input character pattern data etc. and can print them on a recording form. In drawing 79, 8012 is a LBP main part and forms an image in the record paper which is a recording medium based on the character pattern etc. which are supplied. The navigational panel in which, as for 8000, a switch, a LED display for indication, etc. for operation are allotted, and 8001 are printer control units which analyze control of the LBP8012 whole, character pattern information, etc. This printer control unit 8001 mainly changes character pattern information into a video signal, and outputs it to the laser driver 8002. The laser driver 8002 is a circuit for driving the semiconductor laser 8003, and carries out the ON/OFF change of the laser beam 8004 discharged from the semiconductor laser 8003 according to the inputted video signal. The laser beam 8004 is shaken at a horizontal direction in the rotation multiple mirror 8005, and scans the electrostatic drum 8006 top. Thereby, the electrostatic latent image of a character pattern is formed on the electrostatic drum 8006. After this latent image is developed by the development unit 8007 of the electrostatic drum 8006 circumference, it is transferred by the recording form. Using a cut sheet in this recording form, a cut sheet recording form is stored by two or more paper cassettes 8008 corresponding to two or more sorts of papers with which LBP8012 was equipped, is taken in in a device with the paper-feeding roller 8009 and the conveyance rollers 8010 and 8011, and is supplied to the electrostatic drum 8006.

[0153]Drawing 80 is a general-view figure of ink-jet recording device IJRA applicable to the ink-jet printer engine 105 of drawing 1. In the figure, the carriage HC engaged to the spiral slot 9003 of the lead screw 9004 which is interlocked with reciprocal rotation of the drive motor 9011, and is rotated via the driving force transfer gear 9010-9008 has a pin (un-illustrating), and both-way movement is carried out in the arrow a and the direction of b. The inkjet cartridge IJC is carried in this carriage HC. 9001 is a paper control board, continues in the move direction of a carriage and presses paper to the platen 9000. 9006-9007 is a photocoupler and is a home position detection means for checking existence [in this region of the lever 9005 of a carriage], and performing the hand-of-cut change of the motor 9011, etc. 9013 is a member which supports the cap member 9019 which caps the front of a recording head, and 9012 is a suction means which attracts the inside of this cap, and performs suction recovery of a recording head via the opening 9020 in a cap. 9014 is a cleaning blade, 9016 is a member which enables movement of this blade in the direction of order, and these are supported by the main part support plate 9015. It cannot be overemphasized that not this form but a well-known cleaning blade can apply a blade to this example. It is a lever for starting suction of suction recovery, and it moves with movement of the cam 9017 which engages with a carriage, and, as for 9018, move control of the driving force from a drive motor is carried out by publicly known means of communication, such as a clutch change. When a carriage comes to the field by the side of a home position, these capping, cleaning, and suction recovery are constituted so that a request can be processed by operation of the lead screw 9004 in those correspondence positions, but. [suction recovery] If it is made to operate about a request in well-known timing, each is applicable to this example.

[0154]As explained above, according to this embodiment. . [the function of a logic device in which a job is supplied from an information processor] . [the job which held and was supplied to the logic device concerned] Hold the function of the device engine of the logic device control program (the scanning job manager 419, the print job managers 501-509, copy job managers 601-608) to manage and peripheral equipment, and. [the job in the device engine concerned] A job is analyzable using the physical device control program (the scanner controller 420, the LBP controllers 510 and 511, Ink Jet controller 512) to manage.

[0155]Here, a physical device control program is a scanner control program (scanner controller 420) which controls the scanner engine of peripheral equipment. A physical device control program is a laser beam printer control program (LBP controllers 510 and 511) which controls the laser beam printer engine of peripheral equipment. A physical device control program is an ink-jet printer control program (Ink Jet printer controller 512) which controls the ink-jet printer engine of peripheral equipment. [a logic device control program] A print job control program which controls the laser beam printer control program, the ink-jet printer control program or laser beam printer control program, and ink-jet printer control program of peripheral equipment. They are (the print job managers 501-509). A logic device control program is a scanner job control program (scanning job manager 419) which controls the scanner control program of peripheral equipment. [a logic device control program] A copy job control program which controls the scanner control program, the laser beam printer control program, the ink-jet printer control program or laser beam printer control program, and ink-jet printer control program of peripheral equipment. They are (the copy job managers 601-608).

[0156]By such composition, the logic device control program can hold a relation with at least one device engine used for execution of a logic device of a job. The device engine actually used based on the job supplied to the logic device for job execution can be determined. The function which a logic device control program holds can be changed. The job supplied to the logic device can be connected with the job managed by the actually performed physical device, and can be managed. Two or more logic device control programs exist, compare the loading state of two or more logic device control programs, and can choose the optimal logic device control program. It has a generalization program (Supervisor410) which generalizes two or more logic device control programs and the logic device control program of these plurality, The generalization program concerned runs on each logic device control program, and can hold the kind of the subaddress used when supplying a job to said each logic device control program, and connection interface. The list of the information which shows whether a generalization program has an effective or invalid subaddress is held, and the job concerned can be canceled when a job is supplied to the subaddress which is not effective. A generalization program holds the list of an usable user's attestation information, and when the user information included in the inputted job is not included in the list of the attestation information concerned, it can carry out error information of the job concerned. The generalization program can hold the list of the security levels which show the kind of an usable user's attestation judgment processing. The logic device control program can hold the connection interface for transmitting event data to an information processor, and the list of addressee addresses, when an event generates a job in analysis. When an event generates a job in analysis in a logic device control program, event data can be transmitted with reference to the connection interface for transmitting to an information processor, and the list of addressee addresses.

[0157]As explained above, according to this embodiment, according to the function (attribute value) acquired from peripheral equipment, automatic composition (Step 4813) of the user interface of the control program for controlling peripheral equipment can be carried out. According to the acquired function, the display (drawing 81, drawing 82) of the user interface for peripheral equipment is controllable. The information (attribute value) about the setting range of the function of said peripheral equipment is acquirable. The information about a setting range is expressed in the combination (from attribute ID801 to ID805) of the attribute to which a job setup is forbidden. The information (attribute value) about the functional choice of peripheral equipment is acquirable. The attribute list (attribute table) in which the function of peripheral equipment to the peripheral equipment concerned is shown can be acquired, and an attribute value can be acquired by specifying the attribute ID of the attribute list concerned. The attribute list (attribute table) in which each function of the physical device control program of peripheral equipment, a logic device control program, a resource control program, and the generalization control program that generalizes them is shown is acquirable from the peripheral equipment concerned. A physical device control program is a scanner control program which controls the scanner engine of peripheral equipment. A physical device control program is a laser beam printer control program which controls the laser beam printer engine of peripheral equipment.

[0158]A physical device control program is an ink-jet printer control program which controls the ink-jet printer engine of peripheral equipment. .A logic device control program is a print job control program which controls the laser beam printer control program, the ink-jet printer control program or laser beam printer control program, and ink-jet printer control program of peripheral equipment. A logic device control program is a scanner job control program which controls the scanner control program of peripheral equipment. .A logic device control program. [the scanner control program, the laser beam printer control program, the ink-jet printer control program or laser beam printer control program, and ink-jet printer control program of peripheral equipment] It is a copy job control program to control. A resource control program is a font control program which manages the font of peripheral equipment. A resource control program is a form overlay control program which manages the form overlay of peripheral equipment. A resource control program is a log control program which manages the log of peripheral equipment. A resource control program is a color profile control program which manages the color profile of peripheral equipment.

[0159]As explained above, according to this embodiment. [by judging whether based on the function acquired from peripheral equipment (MFT201), a job script can be published to said peripheral equipment (Steps 5402-5404, Steps 5502-5504, Steps 5602-5604)] According to the judgment result, job issue processing is controllable.

[0160]Here A physical device control program of peripheral equipment, a logic device control program, . [each function of a resource control program (the font manager 413, the form overlay manager 414, the log manager 415, the color profile manager 416) and the generalization program which generalizes them] The shown attribute list (attribute table) is acquired from the peripheral equipment concerned. The combination (from attribute ID801 to ID805) of the attribute to which a job setup is forbidden is acquired. A download job or a print job is published to peripheral equipment. An upload job or a scanning job is published to peripheral equipment. A job issuing method is chosen according to the data's attribute (attribute ID601) and for job issue existence position which shows the support of the downloading method of the acquired function. A job issuing method is chosen according to the preservation position of the data an attribute (attribute ID602) and for job issue which shows the support of the uploading method of the acquired function. The data for job issue is print document data. The data for job issue is scanning document data. The data for job issue is font data. The data for job issue is form overlay data. The data for job issue is color profile data. The data for job issue is log data. The data for job issue is font data. The event format data according to the event received from peripheral equipment is acquired, and the event received based on the event format data concerned is analyzed. The log format data of the log data of peripheral equipment and the log data concerned is acquired, and said log data are analyzed based on the log format data concerned. The format data of the billing data (count data) of peripheral equipment and the billing data concerned is acquired, and said billing data is analyzed based on the fee collection format data concerned.

[0161]

[Effect of the Invention]As stated above, according to this invention, with the peripheral equipment connected with the information processor, If an interruption job is inputted from an information processor while performing one of two or more of the device control programs which hold the function of two or more device engines of peripheral equipment, and manage the job in the device engine concerned and processing the job, Other different device control programs from the device control program currently executed can be chosen and executed.

[0162]The device control program can return the execution context of the job which evacuated, after evacuating the execution context of the job which was being performed when an interruption job was inputted and completing execution of this interruption job.

[0163]It can avoid making other interruption jobs input into multiplex during interruption job execution.

[0164]When an interruption copy job is inputted, After a job is supplied to the scanner control program which holds the function of the scanner engine of peripheral equipment and manages the job in the scanning engine concerned, The arbitrary printer control programs which hold the function of the printer engine of peripheral equipment and manage the job in the printer engine concerned can be chosen.

[0165]The discernment information corresponding to the device control program of peripheral equipment, for example, an icon, is displayed in a list, and the user can wedge himself in the display of an information processor by choosing certain one in it, and can publish a job on it. In the icon currently displayed in a list, when an interruption job is published, the icon corresponding to the

device control program chosen and executed is contained.
A user wedges himself by selecting this icon, and a job is published.
The user can know how an interruption job [like] can be published by seeing the information about a device control program.

[Brief Description of the Drawings]
[Drawing 1]It is a figure showing the composition of the multifunctional peripheral equipment (MFT) in which one embodiment of this invention is shown.
[Drawing 2]It is a figure showing the system structure which shows one embodiment of this invention.
[Drawing 3]It is a figure showing the hard structure of the controller of MFT shown in drawing 1.
[Drawing 4]It is a figure showing the soft composition of the controller of MFT shown in drawing 1.
[Drawing 5]It is a figure showing the soft composition of the controller of MFT shown in drawing 1.
[Drawing 6]It is a figure showing the soft composition of the controller of MFT shown in drawing 1.
[Drawing 7]It is a figure showing the attribute table of Supervisor410 shown in drawing 4.
[Drawing 8]It is a figure showing the attribute ID of the attribute table of drawing 7, and the meaning of the model ID.
[Drawing 9]Subaddress: It is a figure showing a service ID conversion table.
[Drawing 10]It is a figure showing the meaning of the connection form ID of drawing 9.
[Drawing 11]Subaddress: It is a figure showing a task type conversion table.
[Drawing 12]It is a figure showing the task type ID meaning of drawing 11.
[Drawing 13]It is a figure showing a user authentication table.
[Drawing 14]It is a figure showing an access control table.
[Drawing 15]It is a figure showing a security level.
[Drawing 16]It is a figure showing an event setting table.
[Drawing 17]It is a figure showing an event format table.
[Drawing 18]It is a figure showing the attribute table of a print manager.
[Drawing 19]It is a figure showing the attribute ID of an attribute table and the meaning of the model ID which are shown in drawing 18.
[Drawing 20]It is a figure showing a job table.
[Drawing 21]It is a figure showing a job request table.
[Drawing 22]It is a figure showing a scanning job manager's attribute table.
[Drawing 23]It is a figure showing the attribute ID of an attribute table and the meaning of the model ID which are shown in drawing 22.
[Drawing 24]It is a figure showing a copy job manager's attribute table.
[Drawing 25]It is a figure showing the attribute ID of an attribute table and the meaning of the model ID which are shown in drawing 24.
[Drawing 26]It is a figure showing a font manager's attribute table.

[Drawing 27]It is a figure showing the attribute ID of an attribute table and the meaning of the model ID which are shown in drawing 26.
[Drawing 28]It is a figure showing a font table.
[Drawing 29]It is a figure showing a form overlay manager's attribute table.
[Drawing 30]It is a figure showing the attribute ID of an attribute table and the meaning of the model ID which are shown in drawing 29.
[Drawing 31]It is a figure showing a form overlay table.
[Drawing 32]It is a figure showing a log manager's attribute table.
[Drawing 33]It is a figure showing the attribute ID of an attribute table and the meaning of the model ID which are shown in drawing 32.
[Drawing 34]It is a figure showing a log table.
[Drawing 35]It is a figure showing the contents of log data.
[Drawing 36]It is a figure showing a log format table.
[Drawing 37]It is a figure showing a color profile manager's attribute table.
[Drawing 38]It is a figure showing the attribute ID of an attribute table and the meaning of the model ID which are shown in drawing 37.
[Drawing 39]It is a figure showing a color profile table.
[Drawing 40]It is a figure showing the attribute table of a printer controller.
[Drawing 41]It is a figure showing the attribute ID of an attribute table and the meaning of the model ID which are shown in drawing 40.
[Drawing 42]It is a figure showing a job queue table.
[Drawing 43]It is a figure showing the status of drawing 42.
[Drawing 44]It is a figure showing the attribute table of a scanner controller.
[Drawing 45]It is a figure showing the attribute ID of an attribute table and the meaning of the model ID which are shown in drawing 44.
[Drawing 46]It is a figure showing the structure of a command packet.
[Drawing 47]It is a flow chart which shows processing of a command packet.
[Drawing 48]It is a flow chart which shows access processing of an attribute table.
[Drawing 49]It is a flow chart of service ID list inquiry processing.
[Drawing 50]It is a flow chart of subaddress inquiry processing.
[Drawing 51]It is a flow chart of inquiry processing of the service ID which specified the task type.
[Drawing 52]It is a figure showing the structure of a job.
[Drawing 53]It is a flow chart of the job script processing by each manager.
[Drawing 54]It is a flow chart of the job script processing by each manager.
[Drawing 55]It is a flow chart of the job processing in a print job manager.
[Drawing 56]It is a flow chart of the job processing in a scanning job manager.
[Drawing 57]It is a flow chart of the job processing in a copy job manager.
[Drawing 58]It is a flow chart of the job processing (download) in a font manager, a form overlay manager, a log manager, and a color profile manager.

[Drawing 59]It is a flow chart of the job processing (upload) in a font manager, a form overlay manager, a log manager, and a color profile manager.

[Drawing 60]It is a flow chart of the job management in each manager.

[Drawing 61]It is a flow chart of event transmission.

[Drawing 62]It is a flow chart of the data (script) transmission from apparatus.

[Drawing 63]It is a figure showing the hard structure of Client PC.

[Drawing 64]It is a figure showing the software (control program) composition of Client PC.

[Drawing 65]It is a flow chart of packet generation / transmitting processing.

[Drawing 66]It is a flow chart of the information acquisition by the side of Client, and software (control program) automatic composition.

[Drawing 67]It is a flow chart of the job issue which specified the task type.

[Drawing 68]It is a figure showing the structure of an event.

[Drawing 69]It is a flow chart of event transmission.

[Drawing 70]It is a flow chart of log processing.

[Drawing 71]It is a flow chart of acquisition and processing of fee collection information.

[Drawing 72]It is a flow chart of job issue (a print job, data download).

[Drawing 73]It is a flow chart of job issue (a scanning job, data upload).

[Drawing 74]It is a flow chart of copy job issue.

[Drawing 75]It is a flow chart of job management command issue.

[Drawing 76]It is a memory map of DISK315 of MFT.

[Drawing 77]It is a memory map of DISK6009 of Client PC.

[Drawing 78]It is a figure showing the composition of Client PC.

[Drawing 79]It is a figure showing LBP applicable to the laser beam printer engine of MFT.

[Drawing 80]It is a figure showing IJRA applicable to the ink-jet printer engine of MFT.

[Drawing 81]It is a figure showing the sample of the display screen displayed on a display.

[Drawing 82]It is a figure showing the sample of the display screen displayed on a display.

[Drawing 83]It is a figure showing a job-scheduling table.

[Drawing 84]It is a figure showing the order job queue table of execution time.

[Drawing 85]It is a figure showing a job control statement text table.

[Drawing 86]It is a figure showing the job processing (2) in a print job manager.

[Drawing 87]It is a figure showing the job processing (3) in a print job manager.

[Drawing 88]It is a figure showing the job processing (4) in a print job manager.

[Drawing 89]It is a figure showing the job processing in an interruption print job manager.

[Drawing 90]It is a figure showing the job processing of a print controller.

[Drawing 91]It is a figure showing the job processing in an interruption copy job manager.

[Drawing 92]It is a flow chart of the job management (2) in each manager.

[Drawing 93]It is a flow chart of the job management (3) in each manager.

[Drawing 94]It is a flow chart of the job management (4) in each manager.

[Drawing 95]It is a flow chart of the job management (5) in each manager.

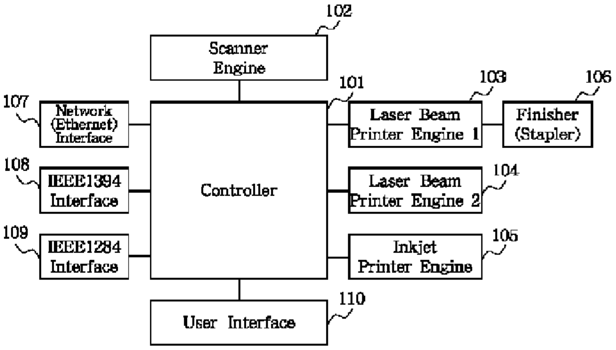
[Drawing 96]It is an option setting dialog at the time of interruption print execution.

[Drawing 97]It is a notice dialog at the time of the end of an interruption print.

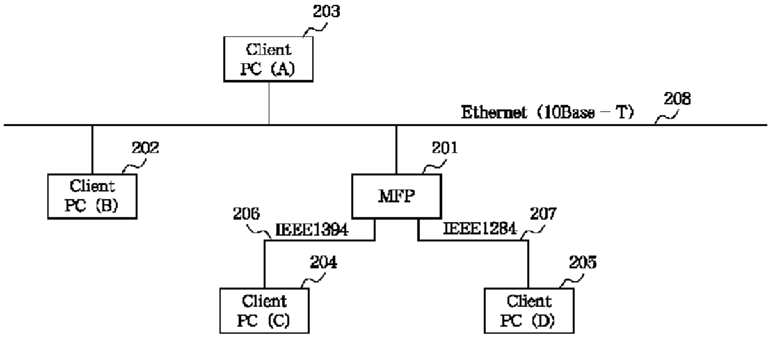
[Drawing 98]It is an option setting dialog at the time of interruption print execution.

[Drawing 99]It is an example of the user interface at the time of interruption print execution.

[Drawing 1]



[Drawing 2]



[Drawing 7]

属性表（スーパーバイザー）

701 702 703

属性ID	型ID	値
10	1	0
11	1	0
100	11	10,11,100,101,102,103,104,105,---
101	11	1,2,3,7,9
102	11	1,3
103	1	1
104	11	1001,1002,1003,1004
105	11	1003
301	3	"SuperMFP－1"
302	13	"Japanese", "English"
303	1	"English"
401	11	45,78,34,13
402	11	565,537,545,523
403	11	100,100,100,200
404	13	0.56,0.78,0.34,0.55
405	3	135.45
406	3	500.00
501	11	200,399,432,234
502	52	Table Data
503	53	Table Data
1001	101	Table Data
1002	102	Table Data
1003	103	Table Data
1004	104	Table Data
2001	11	21,22,23,24

[Drawing 10]

・サービスを受ける時に、どのサブアドレスにジョブを投入したら良いかを記述したもの。

・その接続形態が有効かどうか（使用させるかどうか）もこの表で管理する。

・接続形式ID
0：Internal
1：TCP／IP
2：IEEE1284.4
3：SBB－2

[Drawing 12]

・Service IDがどのような種類のサービスであるかを表す表
・Task TypeはCPCAのタスクタイプに相当
・この表はDispatcherからも参照され、コマンドを各Managerに配信するためにも使用される
・TaskType
0：Supervisor
1：Print
2：Scan
3：Copy
11：Interrupt Print
13：Interrupt Copy
101：Font
102：Form Overlay
103：Log
104：Color Profile

[Drawing 13]

ユーザ認証表

1001 1002 1003

UID	パスワード	管理者
2355	2345	NO
3254	2223	NO
4665	1145	YES
2664	1244	NO

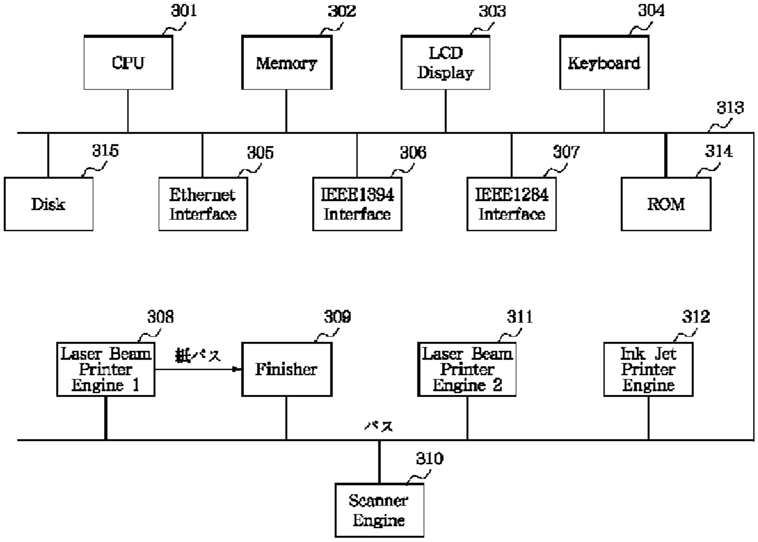
[Drawing 14]

アクセス制御表

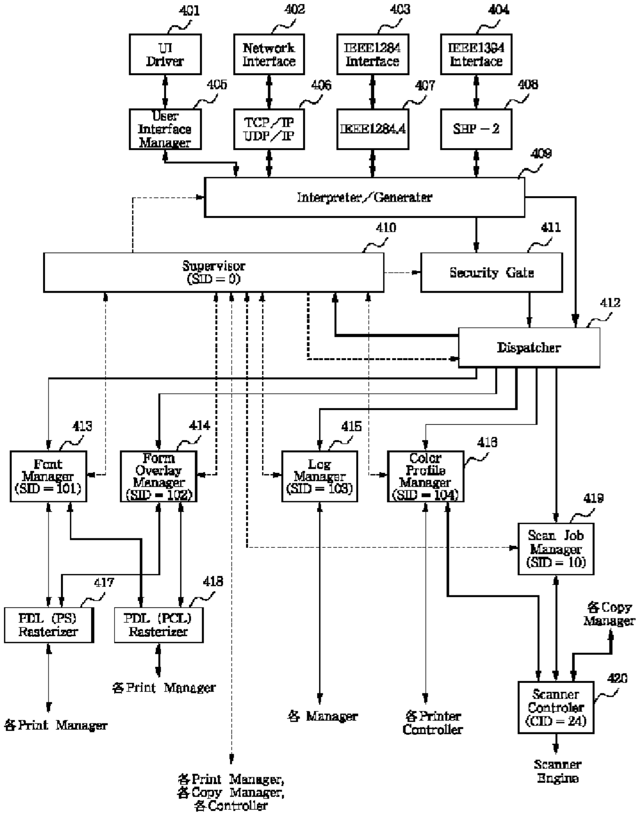
1101 1102 1103

SID	Security Level	UID
0	1	–
1	1	–
2	2	2355,3254
3	3	3254,4665
102	2	3254

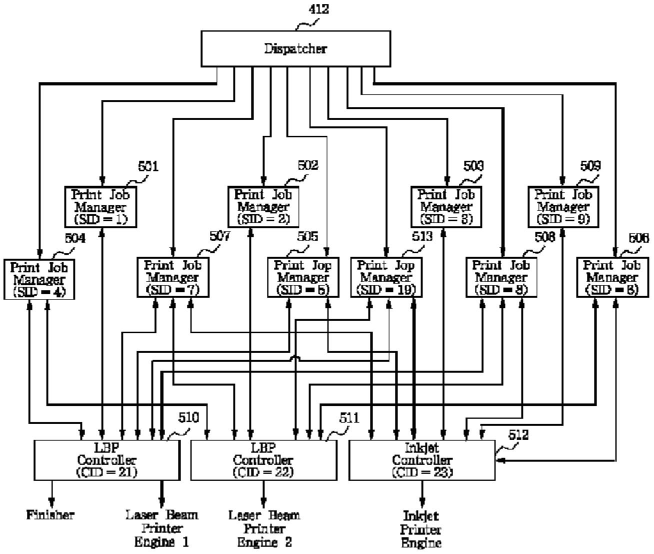
[Drawing 3]



[Drawing 4]



[Drawing 5]


412
Dispatcher
501
Print Job Manager (SID = 1)
502
Print Job Manager (SID = 2)
503
Print Job Manager (SID = 3)
509
Print Job Manager (SID = 9)
504
Print Job Manager (SID = 4)
507
Print Job Manager (SID = 7)
505
Print Jop Manager (SID = 5)
513
Print Jop Manager (SID = 19)
508
Print Job Manager (SID = 8)
506
Print Job Manager (SID = 6)
510
LBP Controller (CID = 21)
511
LBP Controller (CID = 22)
512
Inkjet Controller (CID = 23)
Finisher
Laser Beam Printer Engine 1
Laser Beam Printer Engine 2
Inkjet Printer Engine

[Drawing 15]

- サービス毎のセキュリティレベルを表す
- Security Level
  - 0：何も区別しない
  - 1：管理者のみ区別する
  - 2：認証されたユーザのみ使用可能（認証にUIDのみ使用）
  - 3：認証されたユーザのみ使用可能　（認証にUIDとパスワードを使用）
- 表のUIDの列は、Security Levelが2と3の時のみ使用

[Drawing 16]

イベント設定表（Supervisor）

1201 1202 1203

Event ID	接続形式ID	通知先アドレス
200	0	
399	1	123,222,111,321：9340
432	2	35
234	3	12

[Drawing 17]

イベントフォーマット表

1301 1302

Event ID	イベントフォーマット
200	676,756
399	666
432	698
234	600

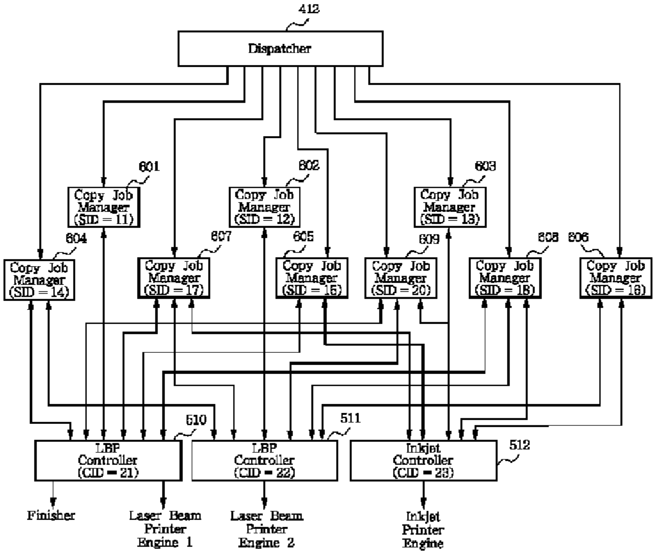
[Drawing 21]

ジョブ依頼表

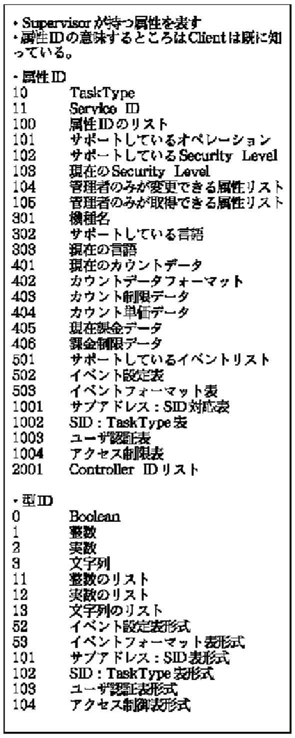
1503 1504 1505

Job ID	Control ID	Job ID
1	21	100
1	22	101
2	21	102
2	22	409
3	–	–

[Drawing 6]


412
Dispatcher
601
Copy Job Manager (SID = 11)
602
Copy Job Manager (SID = 12)
603
Copy Job Manager (SID = 13)
604
Copy Job Manager (SID = 14)
607
Copy Job Manager (SID = 17)
605
Copy Job Manager (SID = 15)
609
Copy Job Manager (SID = 20)
608
Copy Job Manager (SID = 18)
606
Copy Job Manager (SID = 16)
510
LBP Controller (CID = 21)
511
LBP Controller (CID = 22)
512
Inkjet Controller (CID = 23)
Finisher
Laser Beam Printer Engine 1
Laser Beam Printer Engine 2
Inkjet Printer Engine

[Drawing 8]


・Supervisor が持つ属性を表す
・属性IDの意味するところはClientは既に知っている。

・属性ID

10	TaskType
11	Service ID
100	属性IDのリスト
101	サポートしているオペレーション
102	サポートしているSecurity Level
103	現在のSecurity Level
104	管理者のみが変更できる属性リスト
105	管理者のみが取得できる属性リスト
301	機種名
302	サポートしている言語
303	現在の言語
401	現在のカウントデータ
402	カウントデータフォーマット
403	カウント制限データ
404	カウント単価データ
405	現在課金データ
406	課金制限データ
501	サポートしているイベントリスト
502	イベント設定表
503	イベントフォーマット表
1001	サブアドレス：SID対応表
1002	SID：TaskType表
1003	ユーザ認証表
1004	アクセス制限表
2001	Controller IDリスト

・型ID

0	Boolean
1	整数
2	実数
3	文字列
11	整数のリスト
12	実数のリスト
13	文字列のリスト
52	イベント設定表形式
53	イベントフォーマット表形式
101	サブアドレス：SID表形式
102	SID：TaskType表形式
103	ユーザ認証表形式
104	アクセス制御表形式

[Drawing 9]

サブアドレス：サービスID対応表

801	802	803	804	805	806
接続形式ID	サブアドレス	Service ID	有効フラグ	有効ユーザ	無効ユーザ
0	0	0	YES		
0	1	1	YES	2366,5678	
0	2	2	YES		1234,2345
1	9600	0	YES		
1	9601	1	YES		
1	9602	2	YES	1234,2345	
1	8001	101	YES	1234	
1	8002	102	YES		
2	45	0	YES		
2	46	1	YES		
2	47	2	YES		
2	68	8	NO		
2	89	10	YES		
3	23	0	YES		
3	52	1	YES		
3	98	17	YES		

[Drawing 11]

サービスID：タスクタイプ対応表

901	902
Service ID	Task Type
0	0
1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	2
11	3
12	3
13	3
14	3
15	3
16	3
17	3
18	3
19	11
20	13
101	101
102	102
103	103
104	104
21	201
22	201
23	201
24	202

[Drawing 18]

属性表（プリントジョブマネージャ）

1401 1402 1403

属性ID	型ID	値
10	1	1
11	1	4
100	11	10,11,100,101,104,401,…
101	11	11,14,15,17
104	11	1104,1105
401	11	45,78,34,13
402	11	565,537,545,523
403	11	100,100,100,200
404	13	0.56,0.78,0.34,0.55
405	3	135.45
406	3	500.00
501	11	200,399,432,234
502	52	Table Data
601	11	1,2
801	81	4354:3254, 2464, 2456, 4356:124, 535, 6
802	82	3254:1:4324:43143, 2342, 6543, 3245
803	83	5425:2:25432:34324:2:12453, 3424, 2343
804	84	5425:2:25432:4:5425:5:655:2435, 563
805	85	22414:3:4324231:5:43243:2:32144:455
1101	11	3,5
1102	0	NO
1103	11	1,2
1104	1	400
1105	1	100
2001	11	21,22
2002	0	NO
2003	203	Table Data
2004	204	Table Data
2005	205	Table Data
2006	0	YES

[Drawing 20]

ジョブ表

1501 1502 1506 1507 1508

Job ID	Job File Name	実行優先度	指定実行時刻	残存限度時刻
1	P0001	3	13 : 46	14 : 45
2	P1002	3	13 : 46	14 : 45
3	P2001	1	–	–

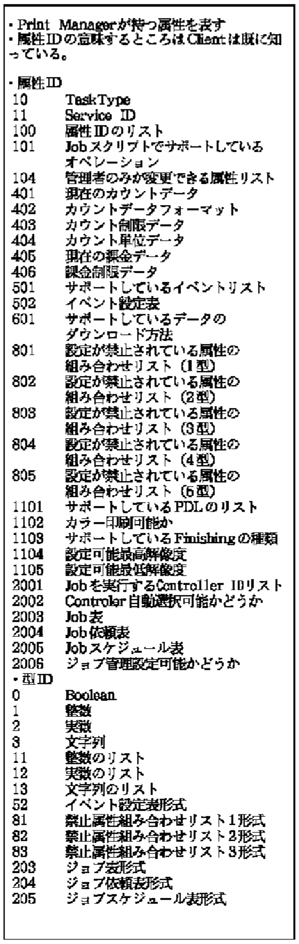
[Drawing 34]

Log表

2301 2302

Log ID	Log Data File名
1	1700
2	1701
3	1702

[Drawing 19]



・Print Managerが持つ属性を表す
・属性IDの意味するところはClientは既に知っている。

・属性ID

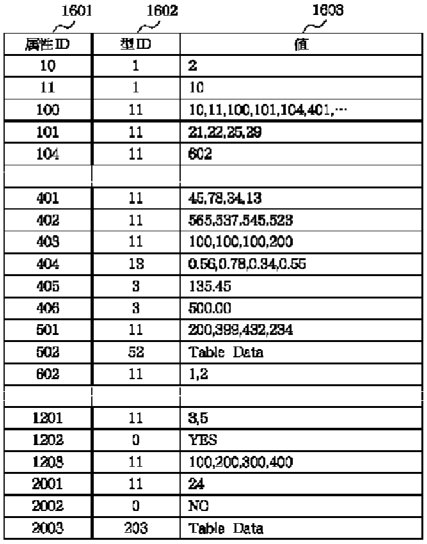
10	TaskType
11	Service ID
100	属性IDのリスト
101	Jobスクリプトでサポートしているオペレーション
104	管理者のみが変更できる属性リスト
401	現在のカウントデータ
402	カウントデータフォーマット
403	カウント制限データ
404	カウント単位データ
405	現在の課金データ
406	課金制限データ
501	サポートしているイベントリスト
502	イベント設定表
601	サポートしているデータのダウンロード方法
801	設定が禁止されている属性の組み合わせリスト（1型）
802	設定が禁止されている属性の組み合わせリスト（2型）
803	設定が禁止されている属性の組み合わせリスト（3型）
804	設定が禁止されている属性の組み合わせリスト（4型）
805	設定が禁止されている属性の組み合わせリスト（5型）
1101	サポートしているPDLのリスト
1102	カラー印刷可能か
1103	サポートしているFinishingの種類
1104	設定可能最高解像度
1105	設定可能最低解像度
2001	Jobを実行するController IDリスト
2002	Controler自動選択可能かどうか
2003	Job表
2004	Job依頼表
2005	Jobスケジュール表
2006	ジョブ管理設定可能かどうか

・型ID

0	Boolean
1	整数
2	実数
3	文字列
11	整数のリスト
12	実数のリスト
13	文字列のリスト
52	イベント設定表形式
81	禁止属性組み合わせリスト1形式
82	禁止属性組み合わせリスト2形式
83	禁止属性組み合わせリスト3形式
203	ジョブ表形式
204	ジョブ依頼表形式
205	ジョブスケジュール表形式

[Drawing 22]

属性表（スキャンジョブマネージャ）


1601
1602
1603

属性ID	型ID	値
10	1	2
11	1	10
100	11	10,11,100,101,104,401,…
101	11	21,22,25,29
104	11	602
401	11	45,78,34,13
402	11	565,537,545,523
403	11	100,100,100,200
404	13	0.56,0.78,0.34,0.55
405	3	135.45
406	3	500.00
501	11	200,399,432,234
502	52	Table Data
602	11	1,2
1201	11	3,5
1202	0	YES
1203	11	100,200,300,400
2001	11	24
2002	0	NO
2003	203	Table Data

[Drawing 23]

・Scan Managerが持つ属性を表す
・属性IDの意味するところはClientは既に知っている。

・属性ID

10	TaskType
11	Service ID
100	属性IDのリスト
101	Jobスクリプトでサポートしているオペレーション
104	管理者のみが変更できる属性リスト
401	現在のカウントデータ
402	カウントデータフォーマット
403	カウント制限データ
404	カウント単価データ
405	現在の課金データ
406	課金制限データ
501	サポートしているイベントリスト
502	イベント設定表
602	サポートしているデータのアップロード方法
1201	サポートしているImage Formatのリスト
1202	カラースキャン可能か
1203	設定可能な解像度のリスト
2001	Jobを実行するController IDリスト
2002	Controller自動選択可能かどうか
2003	Job表

・型ID

0	Boolean
1	整数
2	実数
3	文字列
11	整数のリスト
12	実数のリスト
13	文字列のリスト
52	イベント設定表形式
203	Job表形式

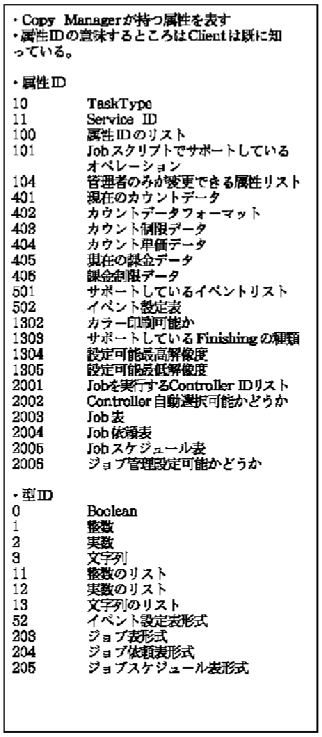
[Drawing 24]

属性表（コピージョブマネージャ）

1701 1702 1703

属性ID	型ID	値
10	1	3
11	1	18
100	11	10,11,100,101,104,401,···
101	11	31,32,35
104	11	1304,1305
401	11	45,78,34,13
402	11	565,537,545,523
403	11	100,100,100,200
404	13	0.56,0.78,0.34,0.55
405	3	135.45
406	3	500.00
501	11	200,399,432,234
502	52	Table Data
1302	0	NO
1303	11	1,2
1304	1	400
1305	1	100
2001	11	21,22,23,24
2002	0	YES
2003	203	Table Data
2004	204	Table Data
2005	203	Table Data
2006	0	YES

[Drawing 25]



・Copy Managerが持つ属性を表す
・属性IDの意味するところはClientは既に知っている。

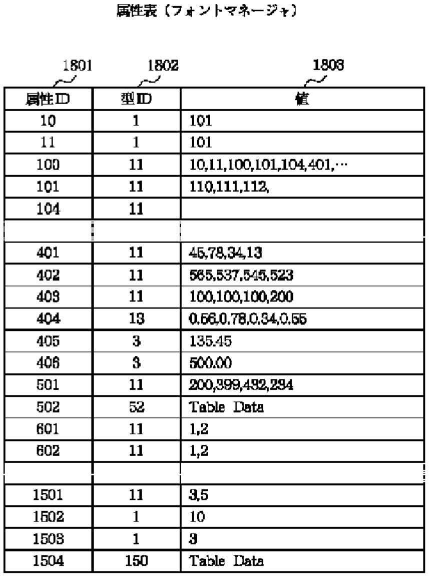
・属性ID

10	TaskType
11	Service ID
100	属性IDのリスト
101	Jobスクリプトでサポートしているオペレーション
104	管理者のみが変更できる属性リスト
401	現在のカウントデータ
402	カウントデータフォーマット
403	カウント制限データ
404	カウント単価データ
405	現在の課金データ
406	課金制限データ
501	サポートしているイベントリスト
502	イベント設定表
1302	カラー印刷可能か
1303	サポートしているFinishingの種類
1304	設定可能最高解像度
1305	設定可能最低解像度
2001	Jobを実行するController IDリスト
2002	Controller自動選択可能かどうか
2003	Job表
2004	Job依頼表
2005	Jobスケジュール表
2006	ジョブ管理設定可能かどうか

・型ID

0	Boolean
1	整数
2	実数
3	文字列
11	整数のリスト
12	実数のリスト
13	文字列のリスト
52	イベント設定表形式
203	ジョブ表形式
204	ジョブ依頼表形式
205	ジョブスケジュール表形式

[Drawing 26]

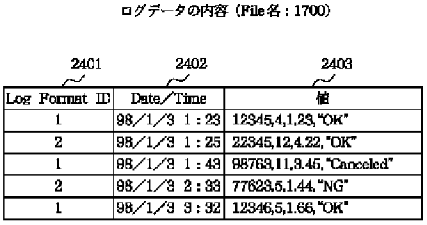


属性表（フォントマネージャ）

1801
1802
1803

属性ID	型ID	値
10	1	101
11	1	101
100	11	10,11,100,101,104,401,…
101	11	110,111,112,
104	11	
401	11	45,78,34,13
402	11	565,537,545,523
403	11	100,100,100,200
404	13	0.56,0.78,0.34,0.55
405	3	135.45
406	3	500.00
501	11	200,399,492,294
502	52	Table Data
601	11	1,2
602	11	1,2
1501	11	3,5
1502	1	10
1503	1	3
1504	150	Table Data

[Drawing 35]

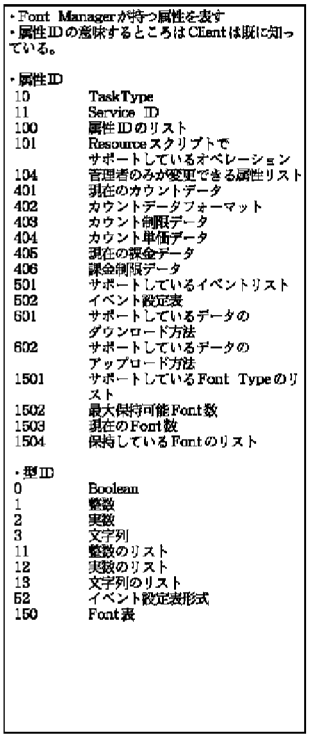


ログデータの内容（File名：1700）

2401
2402
2403

Log Format ID	Date／Time	値
1	98／1／3 1：23	12345,4,1.23, "OK"
2	98／1／3 1：25	22345,12,4.22, "OK"
1	98／1／3 1：43	98763,11,3.45, "Canceled"
2	98／1／3 2：33	77623,5,1.44, "NG"
1	98／1／3 3：32	12346,5,1.66, "OK"

[Drawing 27]



・Font Managerが持つ属性を表す
・属性IDの意味するところはClientは既に知っている。

・属性ID

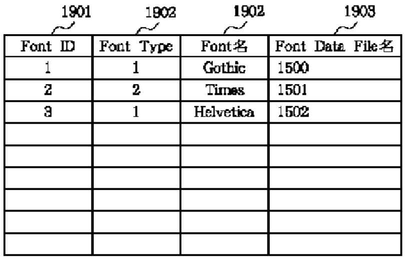
10	TaskType
11	Service ID
100	属性IDのリスト
101	Resourceスクリプトで サポートしているオペレーション
104	管理者のみが変更できる属性リスト
401	現在のカウントデータ
402	カウントデータフォーマット
403	カウント制限データ
404	カウント単価データ
405	現在の課金データ
406	課金制限データ
501	サポートしているイベントリスト
502	イベント設定表
601	サポートしているデータの ダウンロード方法
602	サポートしているデータの アップロード方法
1501	サポートしているFont Typeのリスト
1502	最大保持可能Font数
1503	現在のFont数
1504	保持しているFontのリスト

・型ID

0	Boolean
1	整数
2	実数
3	文字列
11	整数のリスト
12	実数のリスト
13	文字列のリスト
52	イベント設定表形式
150	Font表

[Drawing 28]

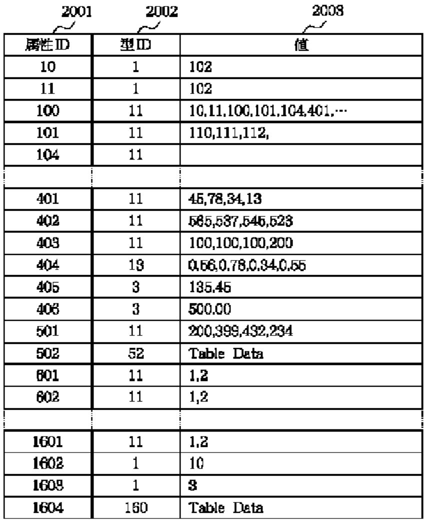
Font表


1901
1902
1902
1903

Font ID	Font Type	Font名	Font Data File名
1	1	Gothic	1500
2	2	Times	1501
3	1	Helvetica	1502

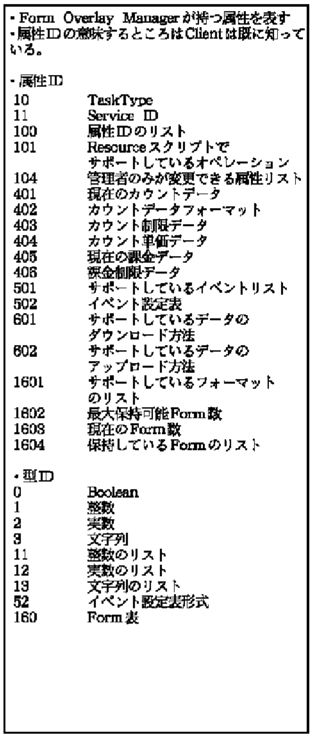
[Drawing 29]

属性表（フォームオーバーレイマネージャ）


2001
2002
2003

属性ID	型ID	値
10	1	102
11	1	102
100	11	10,11,100,101,104,401,···
101	11	110,111,112,
104	11	
401	11	45,78,34,13
402	11	565,537,545,523
403	11	100,100,100,200
404	13	0.56,0.78,0.34,0.55
405	3	135.45
406	3	500.00
501	11	200,399,432,234
502	52	Table Data
601	11	1,2
602	11	1,2
1601	11	1,2
1602	1	10
1603	1	3
1604	150	Table Data

[Drawing 30]



・Form Overlay Managerが持つ属性を表す
・属性IDの意味するところはClientは既に知っている。

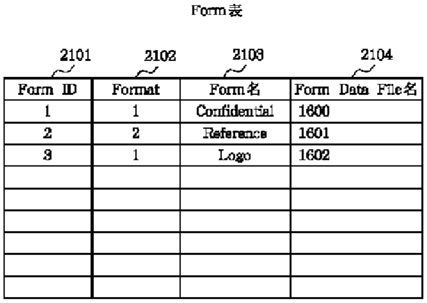
・属性ID

10	TaskType
11	Service ID
100	属性IDのリスト
101	Resourceスクリプトで サポートしているオペレーション
104	管理者のみが変更できる属性リスト
401	現在のカウントデータ
402	カウントデータフォーマット
403	カウント制限データ
404	カウント単価データ
405	現在の課金データ
406	課金制限データ
501	サポートしているイベントリスト
502	イベント設定表
601	サポートしているデータの ダウンロード方法
602	サポートしているデータの アップロード方法
1601	サポートしているフォーマット のリスト
1602	最大保持可能Form数
1603	現在のForm数
1604	保持しているFormのリスト

・型ID

0	Boolean
1	整数
2	実数
3	文字列
11	整数のリスト
12	実数のリスト
13	文字列のリスト
52	イベント設定表形式
160	Form表

[Drawing 31]

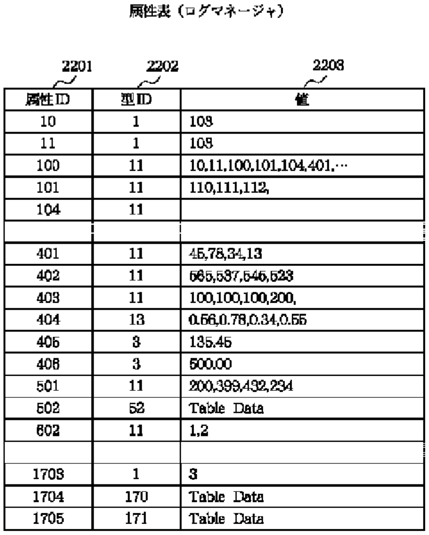


Form表

2101
2102
2103
2104

Form ID	Format	Form名	Form Data File名
1	1	Confidential	1600
2	2	Reference	1601
3	1	Logo	1602

[Drawing 32]



属性表（ログマネージャ）

2201
2202
2203

属性ID	型ID	値
10	1	103
11	1	103
100	11	10,11,100,101,104,401,···
101	11	110,111,112,
104	11	
401	11	45,78,34,13
402	11	565,537,546,523
403	11	100,100,100,200,
404	13	0.56,0.78,0.34,0.55
405	3	135.45
406	3	500.00
501	11	200,399,432,234
502	52	Table Data
602	11	1,2
1703	1	3
1704	170	Table Data
1705	171	Table Data

[Drawing 33]

・Log Managerが持つ属性を表す
・属性IDの意味するところはClientは既に知っている。

・属性ID

10	TaskType
11	Service ID
100	属性IDのリスト
101	Resourceスクリプトで サポートしているオペレーション
104	管理者のみが変更できる属性リスト
401	現在のカウントデータ
402	カウントデータフォーマット
403	カウント制限データ
404	カウント単価データ
405	現在の課金データ
406	課金制限データ
501	サポートしているイベントリスト
502	イベント設定表
602	サポートしているデータの アップロード方法
1703	保持されているLog数
1704	保持しているLogのリスト
1705	Logのフォーマット表

・型ID

0	Boolean
1	整数
2	実数
3	文字列
11	整数のリスト
12	実数のリスト
13	文字列のリスト
52	イベント設定表形式
170	Log表
171	Logフォーマット表形式

[Drawing 36]

ログフォーマット表

2501 2502

Log Format ID	フォーマット
1	1 : 701,1 : 565,1 : 765,1 : 777
2	11 : 701,11 : 565,11 : 765,11 : 777

[Drawing 39]

カラープロファイル表

2701 2702 2703

Profile ID	Format	Profile Data File名
1	3	1800
2	5	1801
3	3	1802

[Drawing 42]

ジョブキュー表

2901 2902 2903

Job ID	Status	File Name
100	1	C100
101	2	C101
102	3	C102
103	3	C103
104	3	C104
105	4	C105

[Drawing 43]

・Printer Controllerで実行中のJobおよび実行待ちのジョブのリスト
・Stausの意味
1：実行終了
2：Enjineで実行中
3：実行待ち
4：実行サスペンド中

[Drawing 37]

属性表（カラープロファイルマネージャ）

2601 2602 2603

属性ID	型ID	値
10	1	104
11	1	104
100	11	10,11,100,101,104,401,···
101	11	110,111,112,
104	11	
401	11	45,78,34,13
402	11	585,537,545,523
403	11	100,100,100,200
404	13	0.56,0.78,0.34,0.55
405	3	135.45
406	3	500.00
501	11	200,399,432,234
502	52	Table Data
601	11	1,2
602	11	1,2
1801	11	3,5
1802	1	10
1803	1	3
1804	180	Table Data

[Drawing 38]

- Color Profile Managerが持つ属性を表す
- 属性IDの意味するところはClientは既に知っている。

- 属性ID

10	TaskType
11	Service ID
100	属性IDのリスト
101	Resourceスクリプトでサポートしているオペレーション
104	管理者のみが変更できる属性リスト
401	現在のカウントデータ
402	カウントデータフォーマット
403	カウント制限データ
404	カウント単価データ
405	現在の課金データ
406	課金制限データ
501	サポートしているイベントリスト
502	イベント設定表
601	サポートしているデータのダウンロード方法
602	サポートしているデータのアップロード方法
1801	サポートしているフォーマットのリスト
1802	最大保持可能Profile数
1803	現在のProfile数
1804	保持しているProfileのリスト

- 型ID

0	Boolean
1	整数
2	実数
3	文字列
11	整数のリスト
12	実数のリスト
13	文字列のリスト
52	イベント設定表形式
180	Profile表

[Drawing 40]

属性表（プリンタコントローラ）

2801 2802 2803

属性ID	型ID	値
12	1	1
13	1	21
100	11	12,13,104…
104	11	
401	11	45,78,34,13
402	11	565,537,545,523
403	11	100,100,100,200
404	13	0.56,0.78,0.34,0.55
405	3	135.45
406	3	500.00
501	11	200,399,432,234
502	52	Table Data
5001	0	NO
5002	11	1,2,3
5003	1	1200
5004	1	100
5005	500	Table Data
5006	0	0
5007	501	Table Data
5008	501	Table Data

[Drawing 41]

- Printer Controllerが持つ属性を表す
- 属性IDの意味するところはClientは既に知っている。

- 属性ID

12	Controller Type
13	Controller ID
100	属性IDのリスト
104	管理者のみが変更できる属性リスト
401	現在のカウントデータ
402	カウントデータフォーマット
403	カウント制限データ
404	カウント単価データ
405	現在の課金データ
406	課金制限データ
501	サポートしているイベントリスト
502	イベント設定表
5001	カラー印刷可能か
5002	サポートしているFinishingの種類
5003	設定可能最高解像度
5004	設定可能最低解像度
5005	Job Que表
5006	割り込みジョブ実行中か
5007	実行中ジョブコンテキスト表
5008	被割り込みジョブコンテキスト表

- 型ID

0	Boolean
1	整数
2	実数
3	文字列
11	整数のリスト
12	実数のリスト
13	文字列のリスト
52	イベント設定表形式
500	Job Que表形式

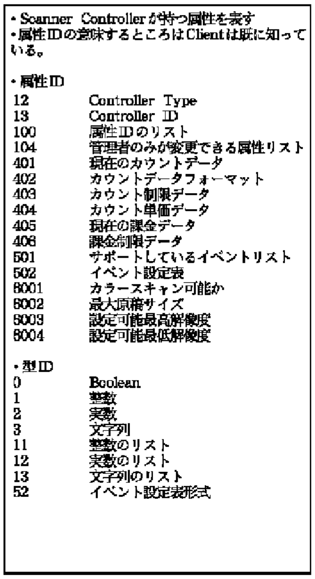
[Drawing 44]

属性表（スキャナコントローラ）

3001 3002 3003

属性ID	型ID	値
12	1	2
13	1	24
100	11	12,13,104,…
104	11	
401	11	45,78,34,13
402	11	565,537,546,523
403	11	100,100,100,200
404	13	0.56,0.78,0.34,0.55
405	3	135.45
406	3	500.00
501	11	200,399,432,234
502	52	Table Data
6001	0	YES
6002	12	200.00,300.00
6003	1	1200
6004	1	100

[Drawing 45]



- Scanner Controller が持つ属性を表す
- 属性IDの意味するところはClientは既に知っている。

- 属性ID

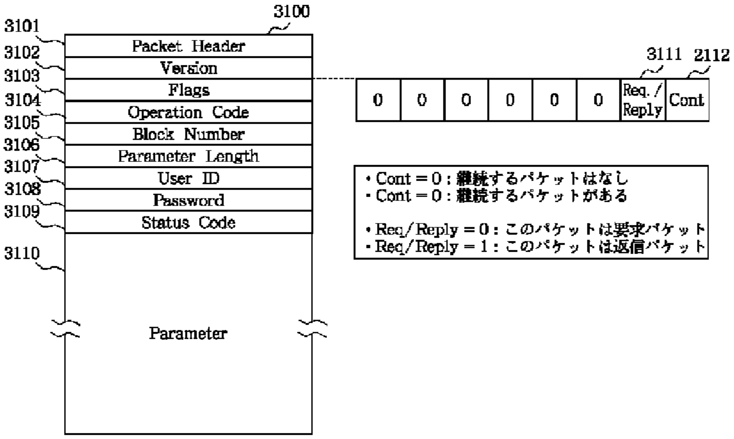
12	Controller Type
13	Controller ID
100	属性IDのリスト
104	管理者のみが変更できる属性リスト
401	現在のカウントデータ
402	カウントデータフォーマット
403	カウント制限データ
404	カウント単価データ
405	現在の課金データ
406	課金制限データ
501	サポートしているイベントリスト
502	イベント設定表
6001	カラースキャン可能か
6002	最大原稿サイズ
6003	設定可能最高解像度
6004	設定可能最低解像度

- 型ID

0	Boolean
1	整数
2	実数
3	文字列
11	整数のリスト
12	実数のリスト
13	文字列のリスト
52	イベント設定表形式

[Drawing 46]

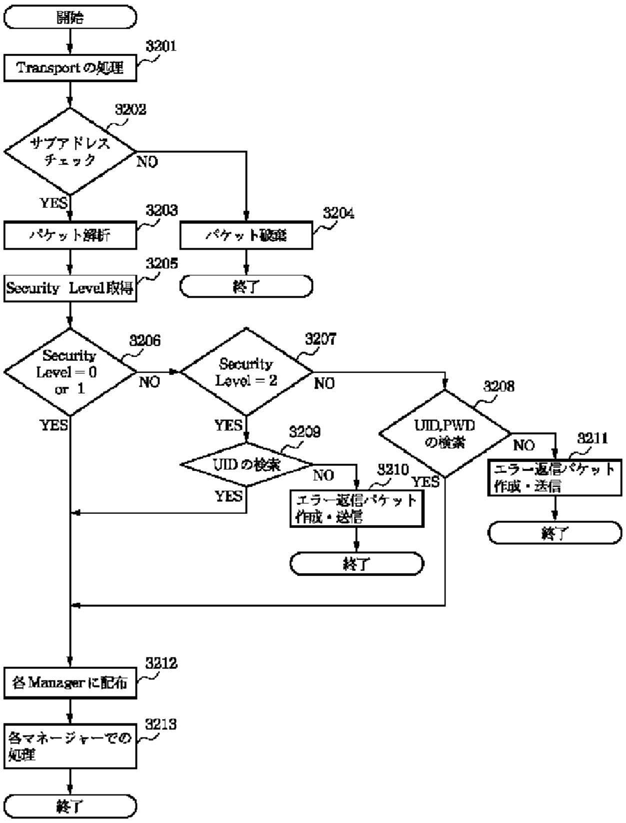
コマンドパケットの構造



- Cont＝0：継続するパケットはなし
- Cont＝0：継続するパケットがある

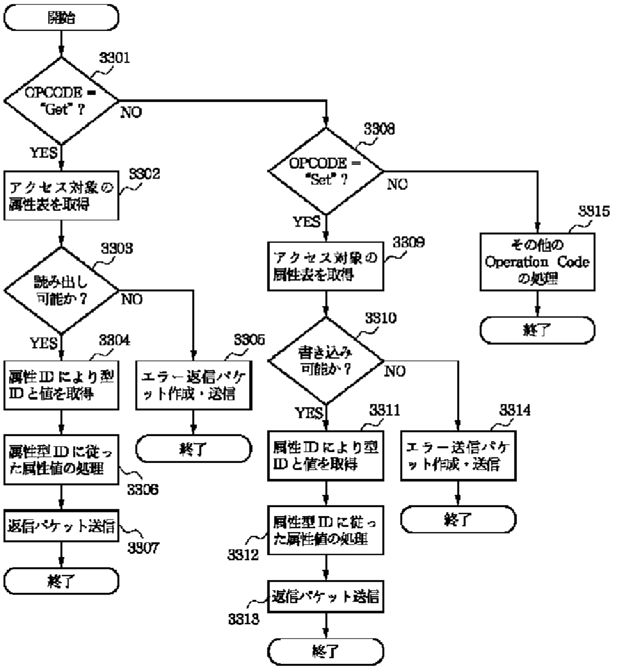
- Req/Reply＝0：このパケットは要求パケット
- Req/Reply＝1：このパケットは返信パケット

[Drawing 47]



コマンドパケットの処理フロー

[Drawing 48]

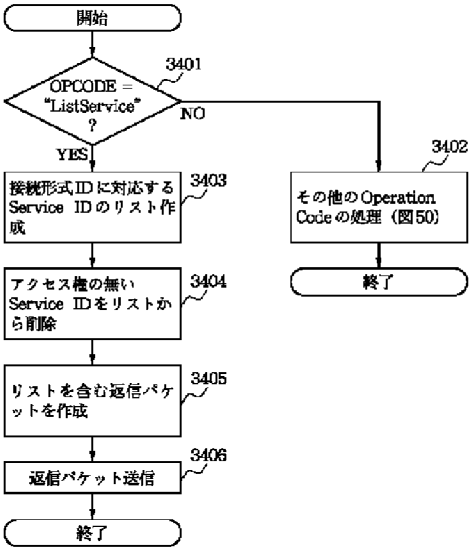


属性表のアクセス処理フロー

開始
3301
OPCODE = "Get"?
NO
YES
3302
アクセス対象の属性表を取得
3303
読み出し可能か?
NO
YES
3304
属性IDにより型IDと値を取得
3305
エラー返信パケット作成・送信
終了
3306
属性型IDに従った属性値の処理
3307
返信パケット送信
終了
3308
OPCODE = "Set"?
NO
YES
3309
アクセス対象の属性表を取得
3310
書き込み可能か?
NO
YES
3311
属性IDにより型IDと値を取得
3312
属性型IDに従った属性値の処理
3313
返信パケット送信
終了
3314
エラー送信パケット作成・送信
終了
3315
その他のOperation Codeの処理
終了

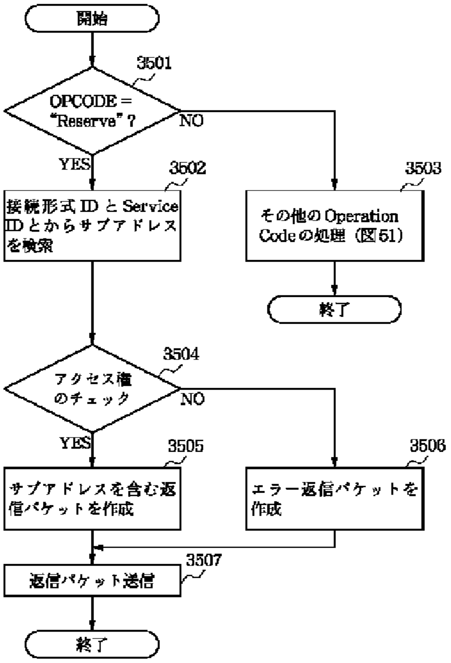
[Drawing 49]

サービスID一覧問合わせ処理

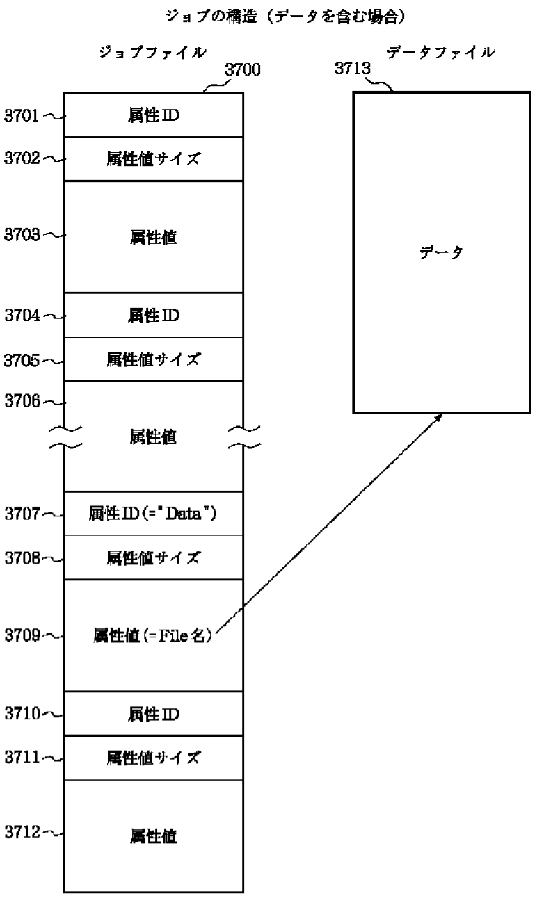

開始
3401
OPCODE = "ListService" ?
NO
YES
3402
その他のOperation Codeの処理（図50）
終了
3403
接続形式IDに対応するService IDのリスト作成
3404
アクセス権の無いService IDをリストから削除
3405
リストを含む返信パケットを作成
3406
返信パケット送信
終了

[Drawing 50]

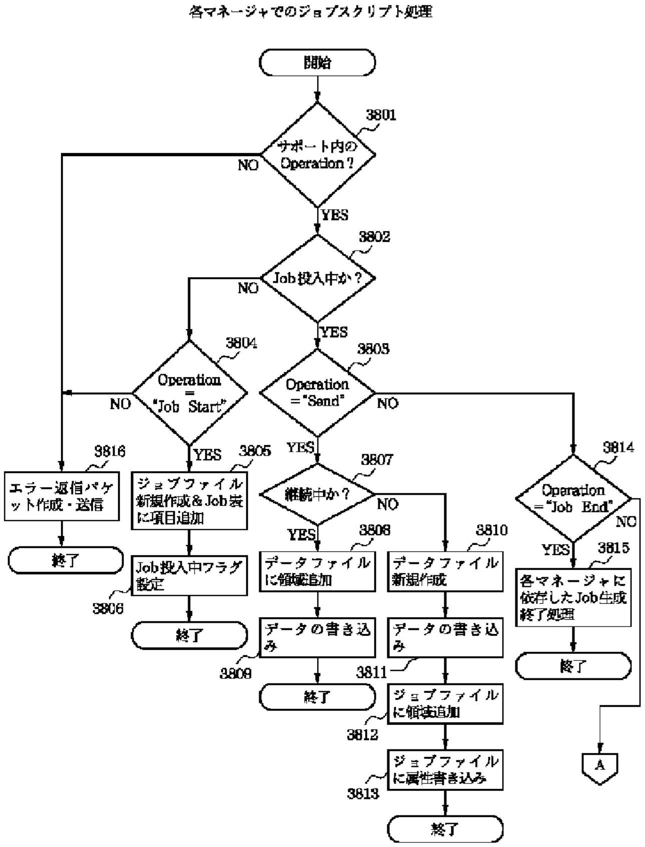
サブアドレス問合わせ処理



[Drawing 52]

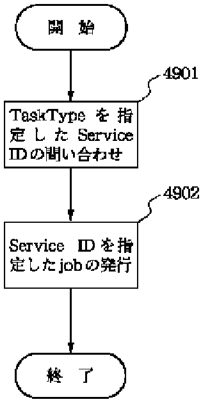

ジョブの構造（データを含む場合）
ジョブファイル
データファイル
3700
3713
3701
属性ID
3702
属性値サイズ
3703
属性値
3704
属性ID
3705
属性値サイズ
3706
属性値
3707
属性ID(="Data")
3708
属性値サイズ
3709
属性値(=File名)
3710
属性ID
3711
属性値サイズ
3712
属性値
データ

[Drawing 53]


各マネージャでのジョブスクリプト処理
開始
3801
サポート内のOperation？
NO
YES
3802
Job投入中か？
NO
YES
3804
Operation = "Job Start"
NO
YES
3803
Operation = "Send"
NO
YES
3816
エラー返信パケット作成・送信
終了
3805
ジョブファイル新規作成＆Job表に項目追加
3806
Job投入中フラグ設定
終了
3807
継続中か？
NO
YES
3808
データファイルに領域追加
3809
データの書き込み
終了
3810
データファイル新規作成
3811
データの書き込み
3812
ジョブファイルに領域追加
3813
ジョブファイルに属性書き込み
終了
3814
Operation = "Job End"
NO
YES
3815
各マネージャに依存したJob生成終了処理
終了
A

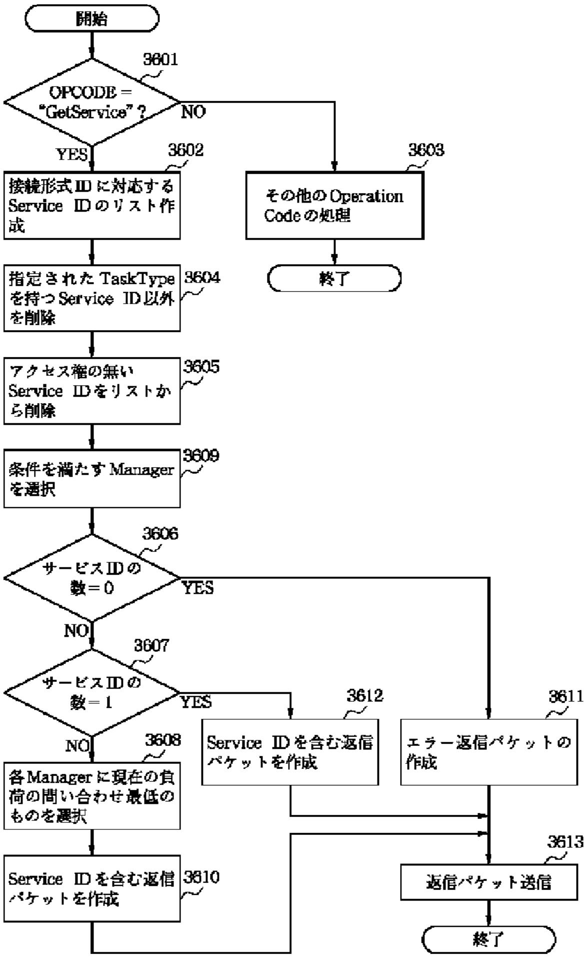
[Drawing 67]

タスクタイプを指定したジョブの発行

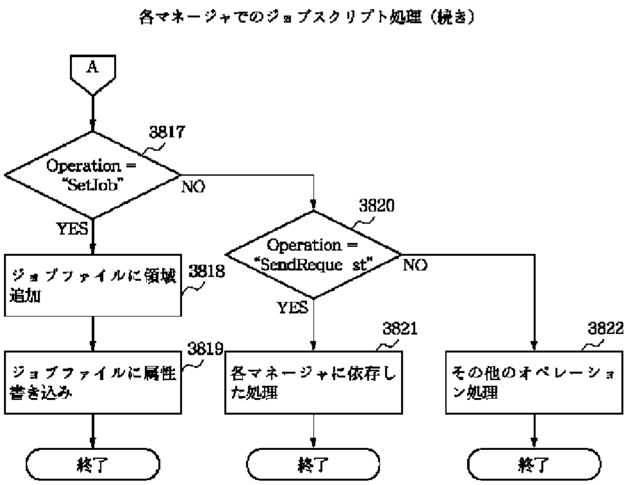

開　始
4901
TaskType を指定した Service IDの問い合わせ
4902
Service ID を指定した job の発行
終　了

[Drawing 51]

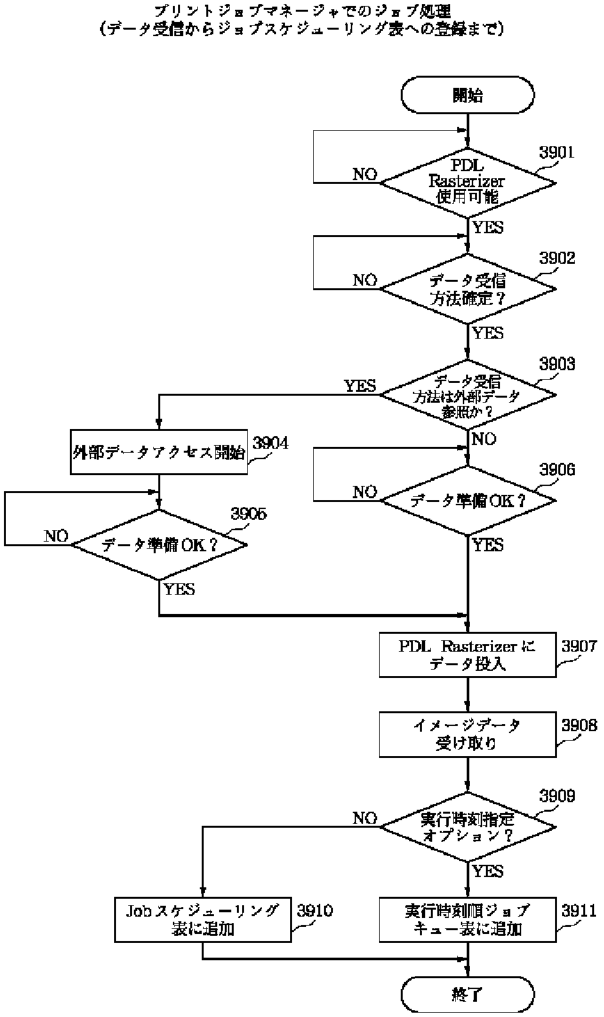
タスクタイプを指定したサービスIDの問い合わせ処理


開始
3601
OPCODE = "GetService"？
NO
YES
3602
接続形式IDに対応するService IDのリスト作成
3603
その他のOperation Codeの処理
終了
指定されたTaskTypeを持つService ID以外を削除
3604
アクセス権の無いService IDをリストから削除
3605
条件を満たすManagerを選択
3609
3606
サービスIDの数＝0
YES
NO
3607
サービスIDの数＝1
YES
3612
3611
NO
3608
Service IDを含む返信パケットを作成
エラー返信パケットの作成
各Managerに現在の負荷の問い合わせ最低のものを選択
3613
Service IDを含む返信パケットを作成
3610
返信パケット送信
終了

[Drawing 54]

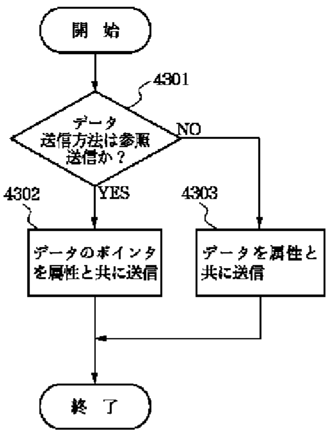

各マネージャでのジョブスクリプト処理（続き）
A
3817
Operation = "SetJob"
NO
YES
3820
Operation = "SendReque st"
NO
YES
ジョブファイルに領域追加
3818
ジョブファイルに属性書き込み
3819
3821
各マネージャに依存した処理
3822
その他のオペレーション処理
終了
終了
終了

[Drawing 55]



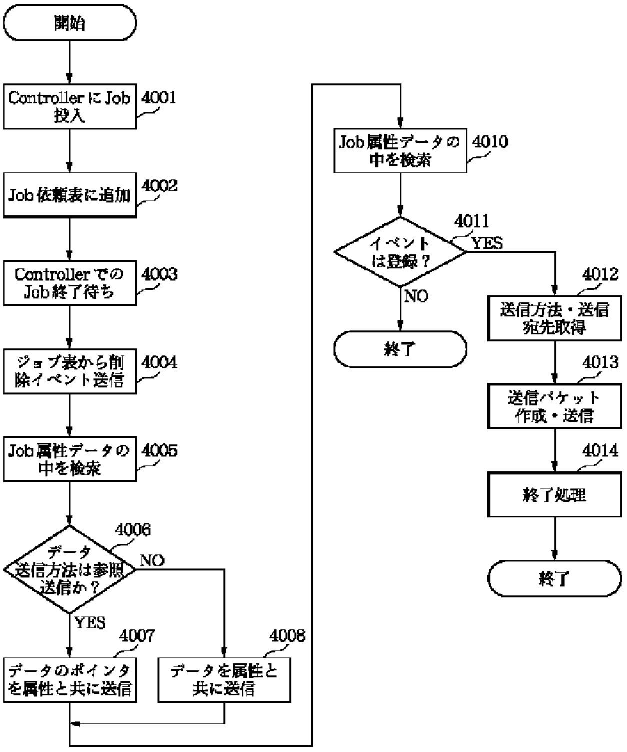
[Drawing 59]

フォントマネージャ，フォームオーバーレイマネージャ，ログマネージャ，
カラープロファイルマネージャでのジョブ処理
（アップロード）



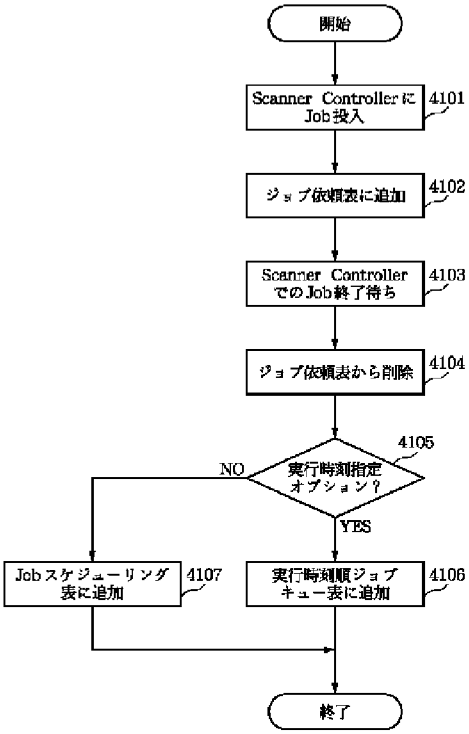
[Drawing 56]

スキャンジョブマネージャでのジョブ処理



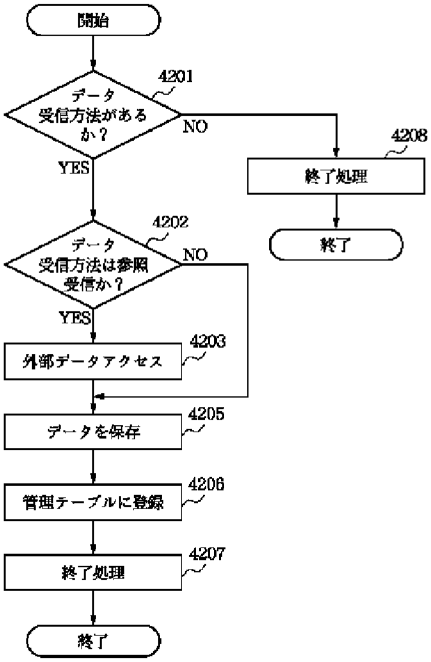
[Drawing 57]

コピージョブマネージャでのジョブ処理

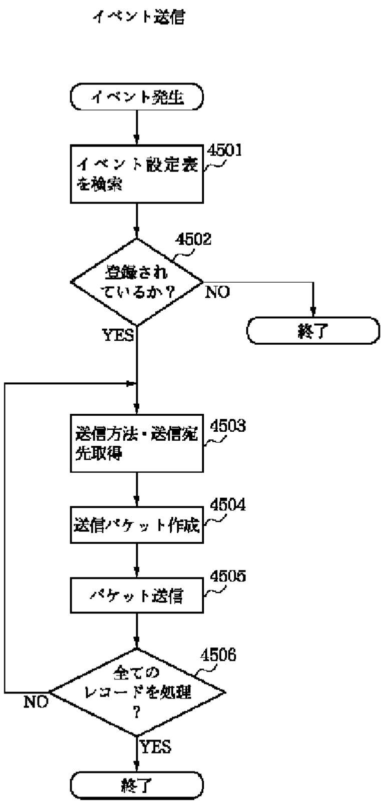


[Drawing 58]

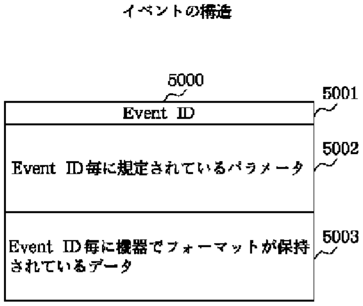
フォントマネージャ、フォームーオーバーレイマネージャ、ログマネージャ、
カラープロファイルマネージャでのジョブ処理（ダウンロード）



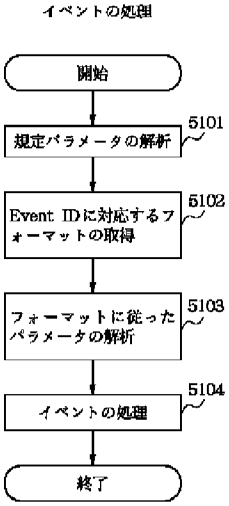
[Drawing 61]



[Drawing 68]

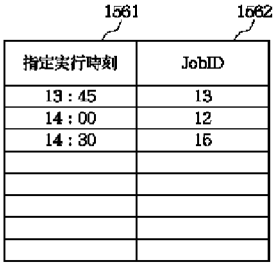

イベントの構造
5000
Event ID
5001
Event ID毎に規定されているパラメータ
5002
Event ID毎に機器でフォーマットが保持されているデータ
5003

[Drawing 69]


イベントの処理
開始
規定パラメータの解析
5101
Event IDに対応するフォーマットの取得
5102
フォーマットに従ったパラメータの解析
5103
イベントの処理
5104
終了

[Drawing 84]

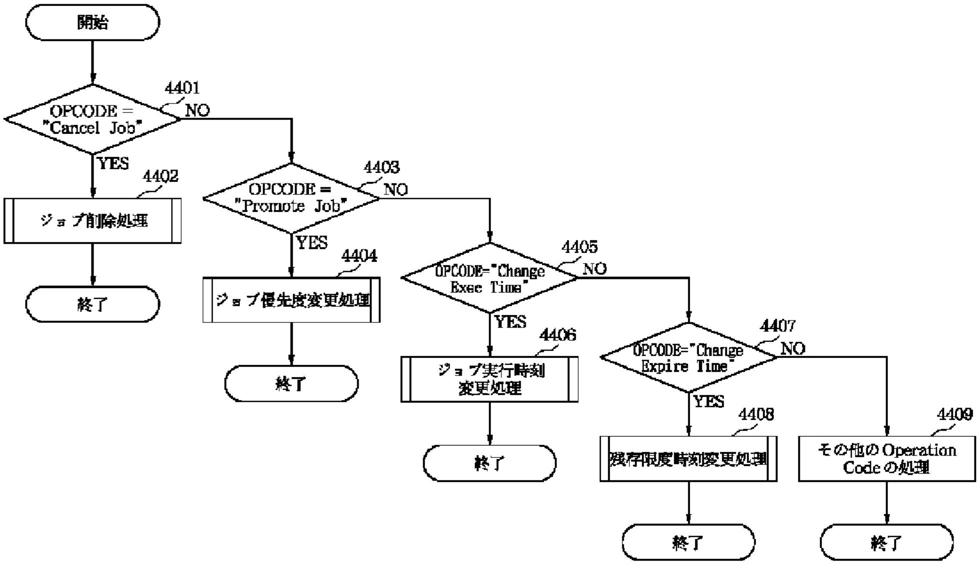
実行時刻順ジョブキュー表



指定実行時刻	JobID
13 : 45	13
14 : 00	12
14 : 30	15

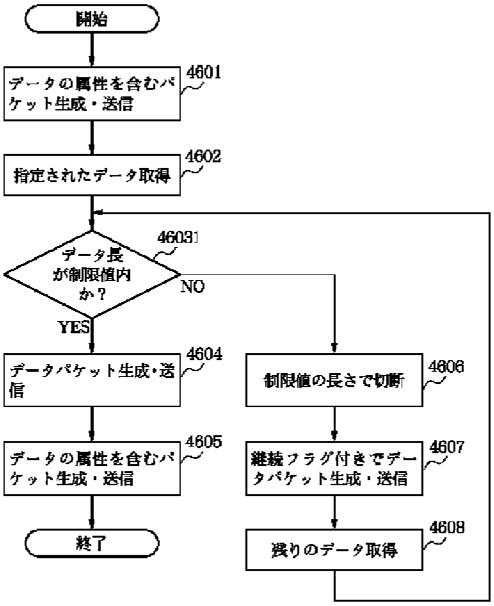
[Drawing 60]

各マネージャでのジョブ管理

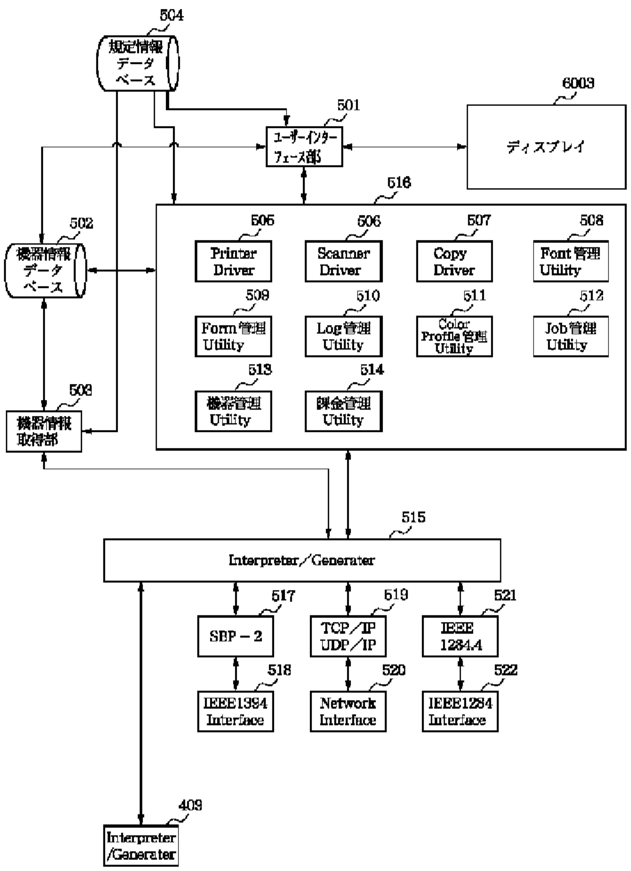


[Drawing 62]

機器からのデータ（スクリプト）送信

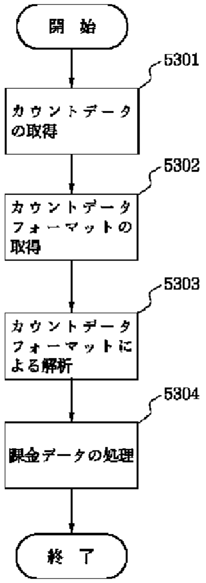


[Drawing 64]

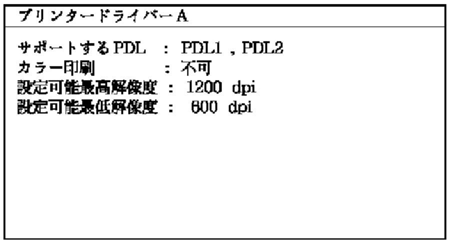


[Drawing 71]

課金情報の取得・処理


開　始
5301
カウントデータの取得
5302
カウントデータフォーマットの取得
5303
カウントデータフォーマットによる解析
5304
課金データの処理
終　了

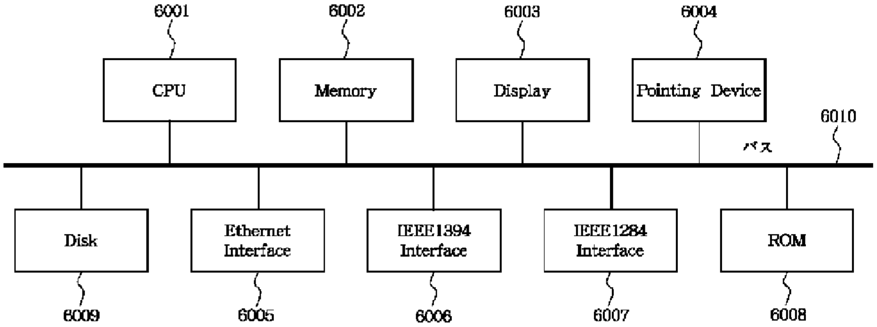
[Drawing 81]



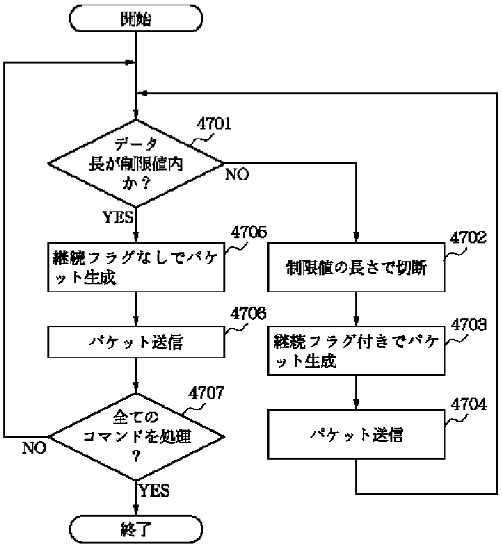
プリンタードライバーA

サポートするPDL　:　PDL1 , PDL2
カラー印刷　:　不可
設定可能最高解像度　:　1200 dpi
設定可能最低解像度　:　600 dpi

[Drawing 63]



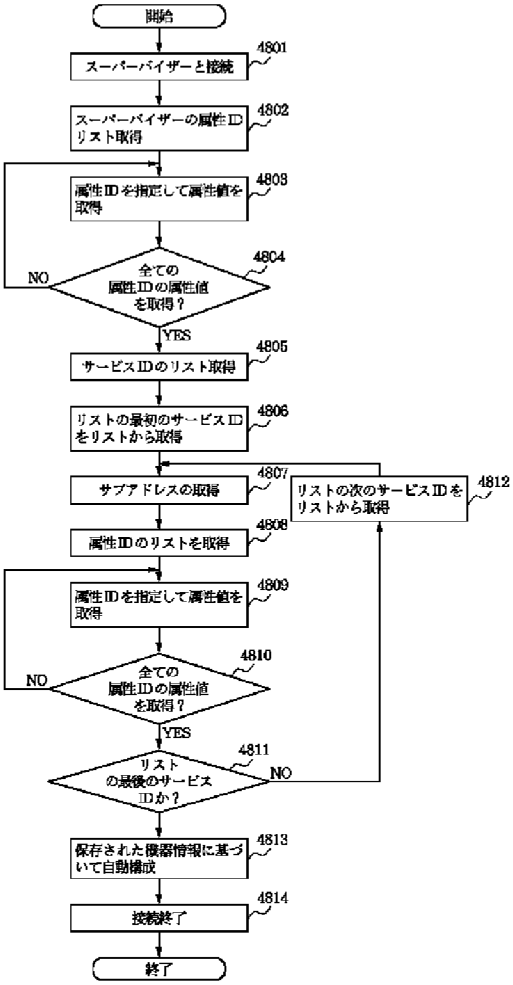
[Drawing 65]



パケット生成・送信処理

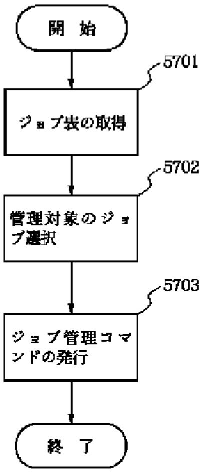
[Drawing 66]

クライアント側での情報取得およびソフトウエア自動構成


開始
スーパーバイザーと接続
4801
スーパーバイザーの属性IDリスト取得
4802
属性IDを指定して属性値を取得
4803
全ての属性IDの属性値を取得？
4804
NO
YES
サービスIDのリスト取得
4805
リストの最初のサービスIDをリストから取得
4806
サブアドレスの取得
4807
リストの次のサービスIDをリストから取得
4812
属性IDのリストを取得
4808
属性IDを指定して属性値を取得
4809
全ての属性IDの属性値を取得？
4810
NO
YES
リストの最後のサービスIDか？
4811
NO
保存された機器情報に基づいて自動構成
4813
接続終了
4814
終了

[Drawing 75]

ジョブ管理コマンドの発行



[Drawing 82]

プリンタードライバーB
サポートするPDL ： PDL3 カラー印刷 ： 可 設定可能最高解像度 ： 600 dpi 設定可能最低解像度 ： 300 dpi

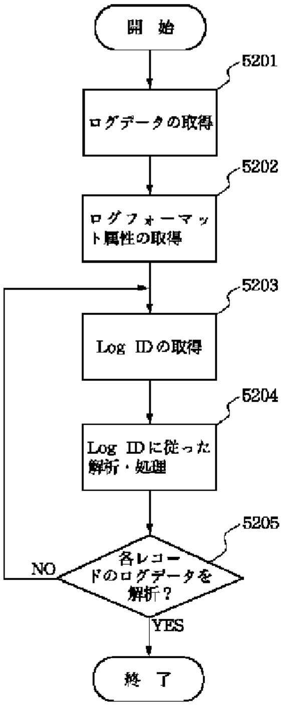
[Drawing 83]

ジョブスケジューリング表

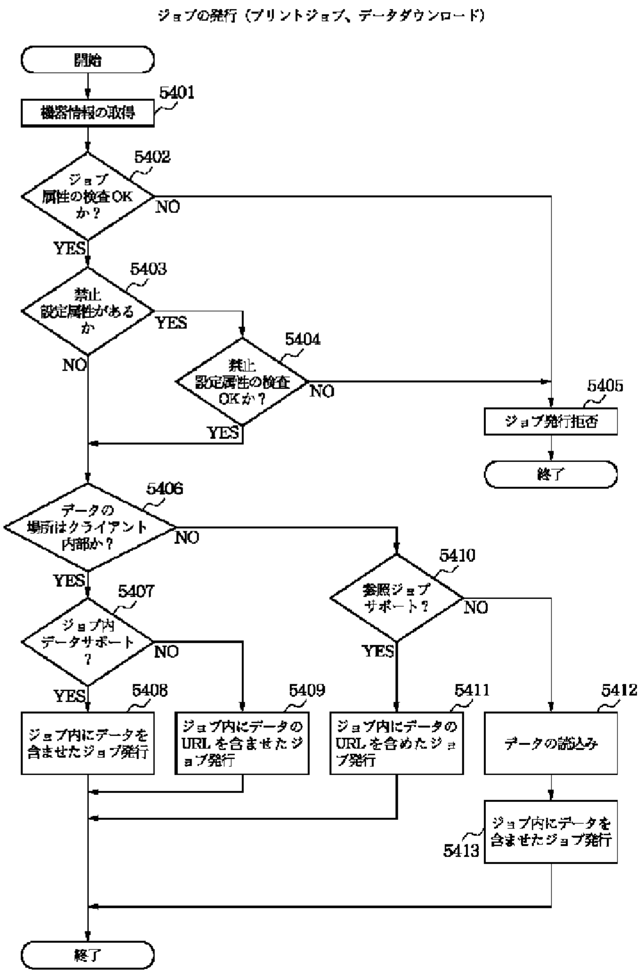
1551	優先度１のJobID	5	7	8		
1552	優先度２のJobID	2	3	4		
1553	優先度３のJobID	1	6			

[Drawing 70]

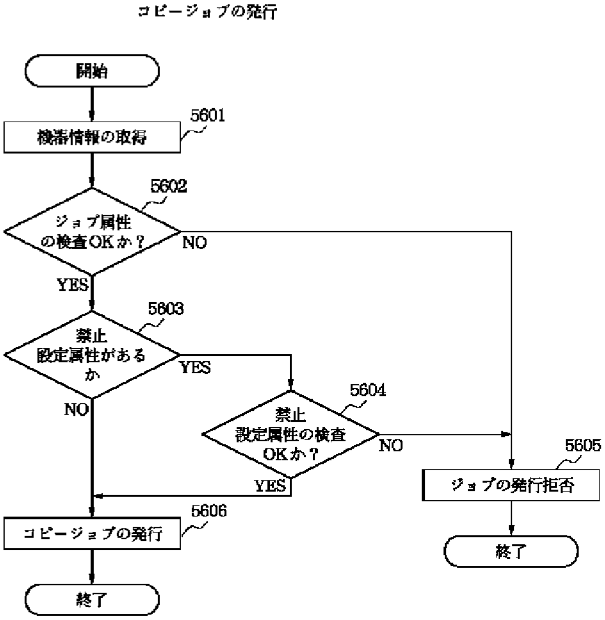
ログの処理



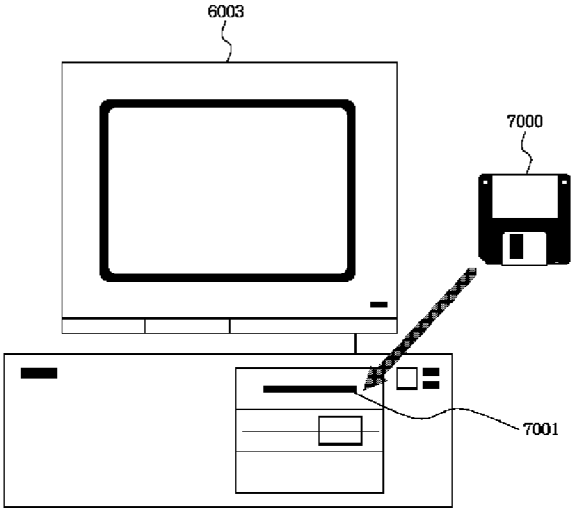
[Drawing 72]



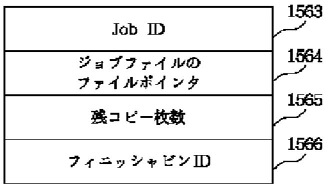
[Drawing 74]



[Drawing 78]

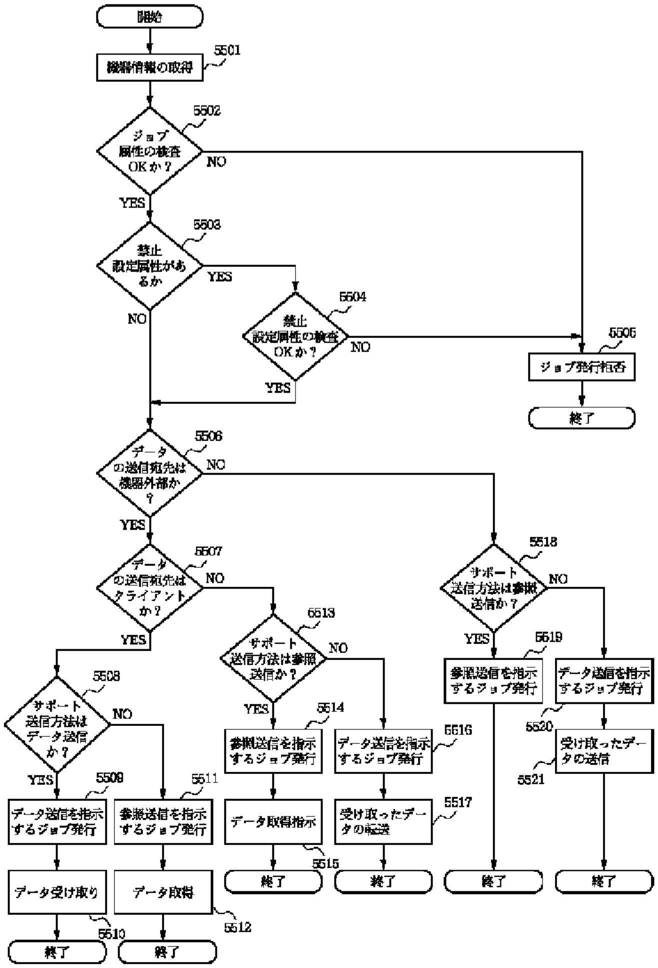


[Drawing 85]

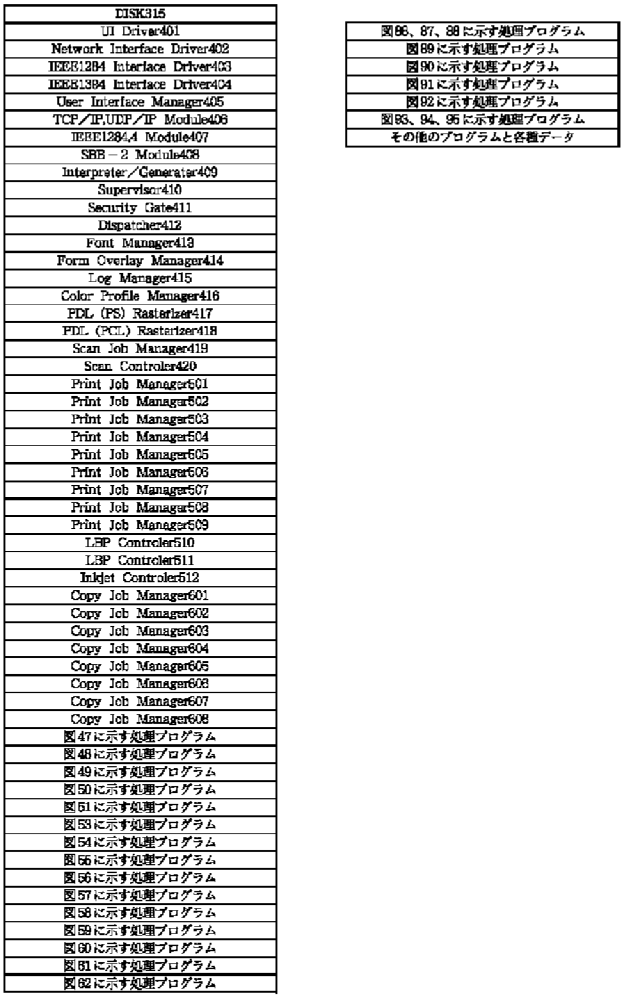

ジョブコンテキスト表

[Drawing 73]

ジョブの発行（スキャンジョブ、データアップロード）



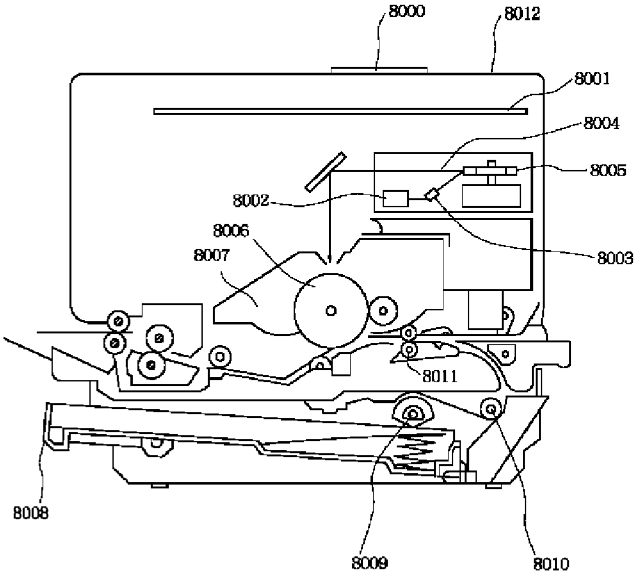
[Drawing 76]



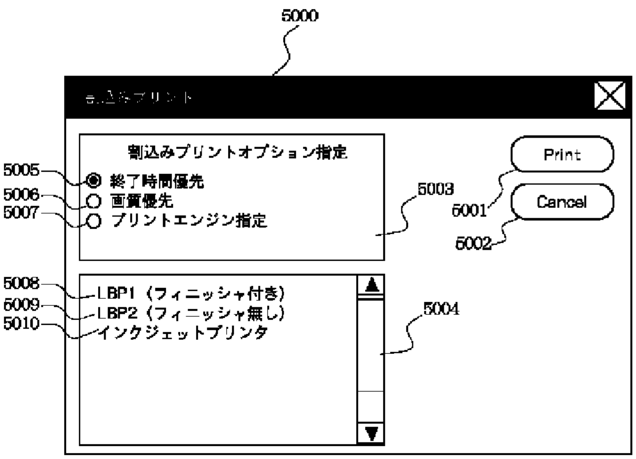
DISK315
UI Driver401
Network Interface Driver402
IEEE1284 Interface Driver403
IEEE1394 Interface Driver404
User Interface Manager405
TCP／IP,UDP／IP Module406
IEEE1284.4 Module407
SBB－2 Module408
Interpreter／Generater409
Supervisor410
Security Gate411
Dispatcher412
Font Manager413
Form Overlay Manager414
Log Manager415
Color Profile Manager416
PDL (PS) Rasterizer417
PDL (PCL) Rasterizer418
Scan Job Manager419
Scan Controler420
Print Job Manager501
Print Job Manager502
Print Job Manager503
Print Job Manager504
Print Job Manager505
Print Job Manager506
Print Job Manager507
Print Job Manager508
Print Job Manager509
LBP Controler510
LBP Controler511
Inkjet Controler512
Copy Job Manager601
Copy Job Manager602
Copy Job Manager603
Copy Job Manager604
Copy Job Manager605
Copy Job Manager606
Copy Job Manager607
Copy Job Manager608
図47に示す処理プログラム
図48に示す処理プログラム
図49に示す処理プログラム
図50に示す処理プログラム
図51に示す処理プログラム
図53に示す処理プログラム
図54に示す処理プログラム
図55に示す処理プログラム
図56に示す処理プログラム
図57に示す処理プログラム
図58に示す処理プログラム
図59に示す処理プログラム
図60に示す処理プログラム
図61に示す処理プログラム
図62に示す処理プログラム

図86、87、88に示す処理プログラム
図89に示す処理プログラム
図90に示す処理プログラム
図91に示す処理プログラム
図92に示す処理プログラム
図93、94、95に示す処理プログラム
その他のプログラムと各種データ

[Drawing 79]



[Drawing 96]



割込みプリント

割込みプリントオプション指定

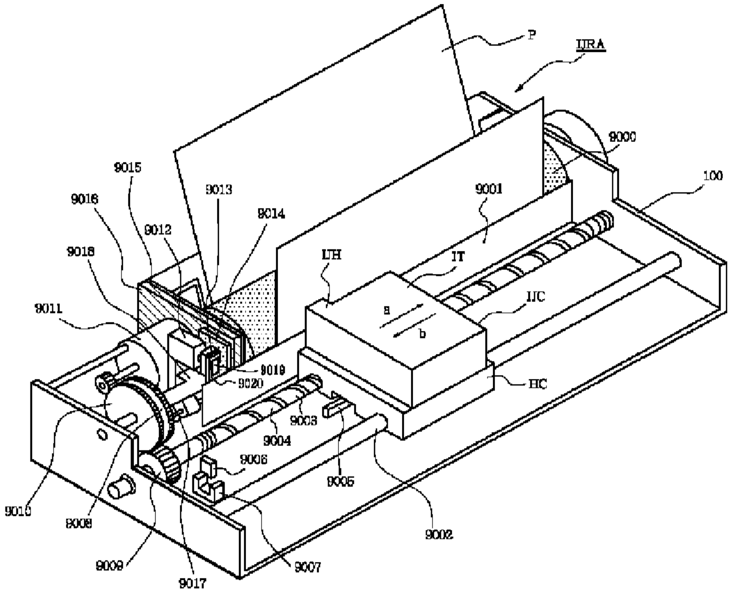
● 終了時間優先
○ 画質優先
○ プリントエンジン指定

LBP1（フィニッシャ付き）
LBP2（フィニッシャ無し）
インクジェットプリンタ

[Drawing 77]

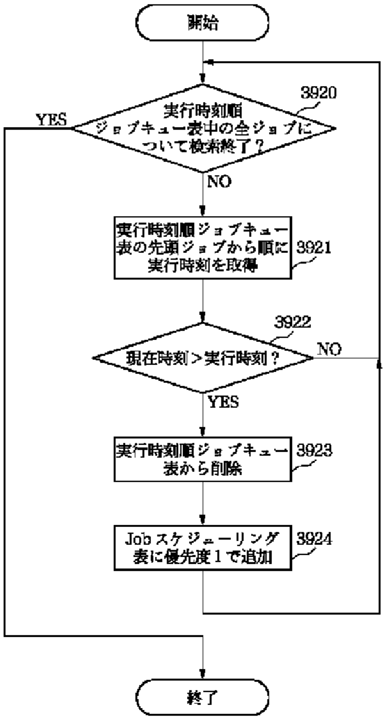
DISK6009
規定情報データベース504
機器情報データベース502
ユーザーインターフェースプログラム501
機器情報取得プログラム503
Printer Driver505
Scanner Driver506
Copy Driver507
Font管理Utility508
Form管理Utility509
Log管理Utility510
Color Profile管理Utility511
Job管理Utility512
機器管理Utility513
課金管理Utility514
Interpreter／Generater515
SBB－2 Module517
IEEE1394 Interface Driver518
TCP／IP,UDP／IP Module519
Network Interface Driver520
IEEE1284.4 Module521
IEEE1284.4Interface Driver522
図65に示すプログラム
図66に示すプログラム
図67に示すプログラム
図69に示すプログラム
図70に示すプログラム
図71に示すプログラム
図72に示すプログラム
図73に示すプログラム
図74に示すプログラム
図75に示すプログラム
Operation System
ドキュメント編集プログラム
その他のプログラムと各種データ

[Drawing 80]



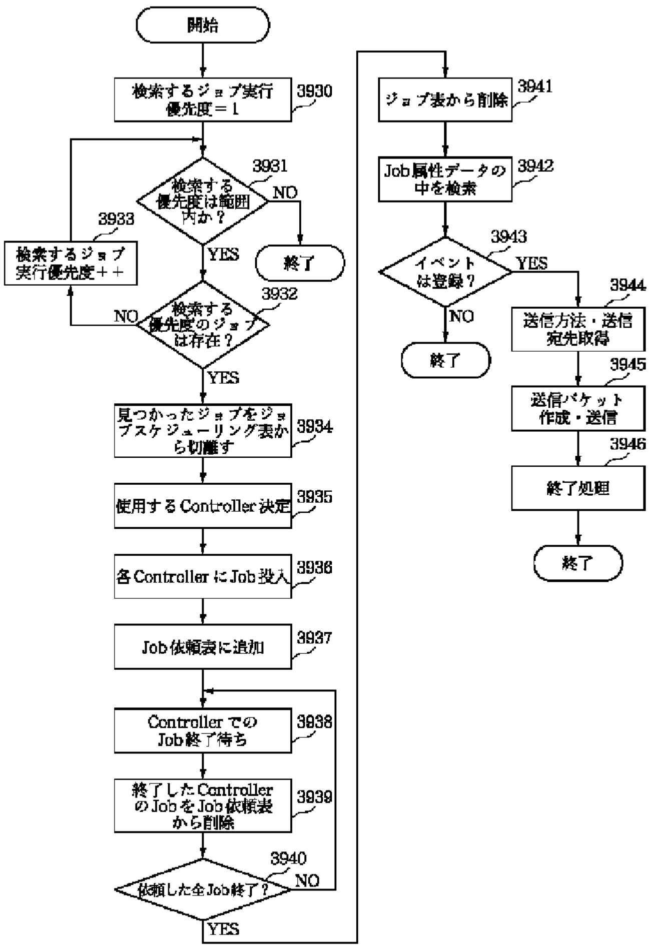
[Drawing 86]

プリントジョブマネージャでのジョブ処理
(実行時刻順ジョブキュー表からジョブスケジューリング表への登録)



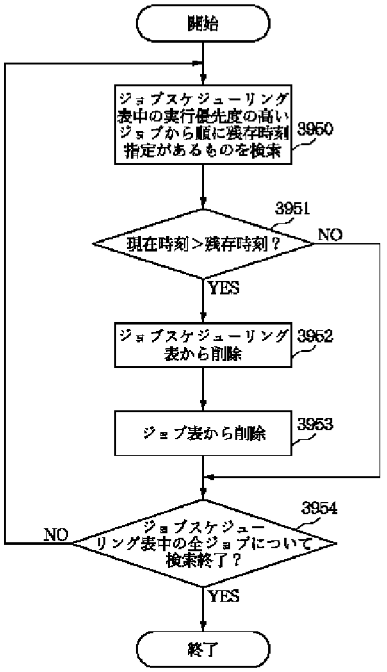
[Drawing 87]

プリントジョブマネージャでのジョブ処理
（ジョブスケジューリング表から実行終了まで）



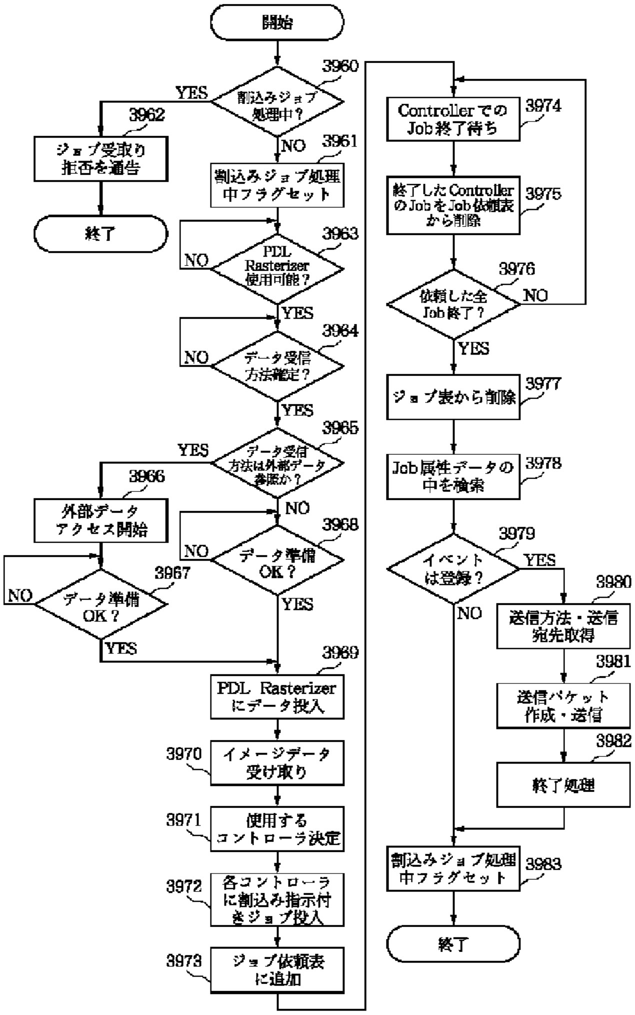
[Drawing 88]

プリントジョブマネージャでのジョブ処理
（残存限度時刻指定のあるジョブの削除）


開始
ジョブスケジューリング表中の実行優先度の高いジョブから順に残存時刻指定があるものを検索
3950
現在時刻＞残存時刻？
3951
NO
YES
ジョブスケジューリング表から削除
3952
ジョブ表から削除
3953
ジョブスケジューリング表中の全ジョブについて検索終了？
3954
NO
YES
終了

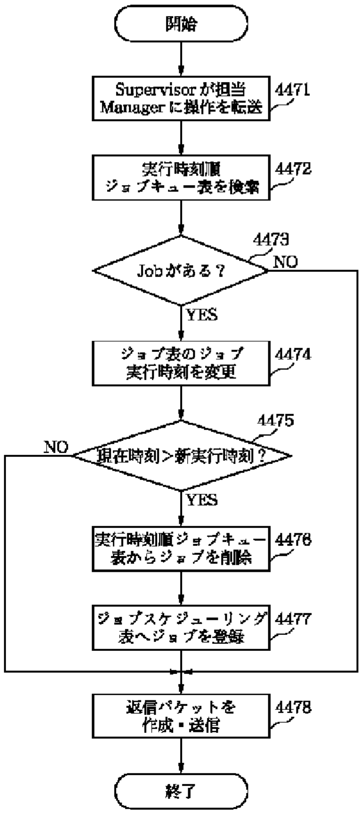
[Drawing 89]

割り込みプリントジョブマネージャでのジョブ処理

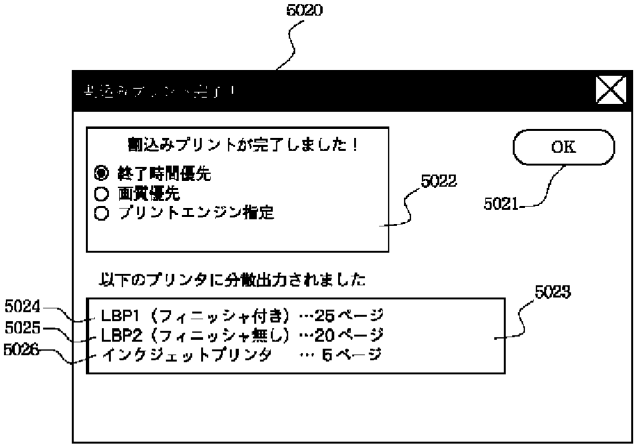


[Drawing 94]

ジョブ管理
（実行時刻変更処理）


開始
Supervisor が担当 Manager に操作を転送
4471
実行時刻順ジョブキュー表を検索
4472
Job がある？
4473
NO
YES
ジョブ表のジョブ実行時刻を変更
4474
現在時刻＞新実行時刻？
4475
NO
YES
実行時刻順ジョブキュー表からジョブを削除
4476
ジョブスケジューリング表へジョブを登録
4477
返信パケットを作成・送信
4478
終了

[Drawing 97]



割込みプリント完了！

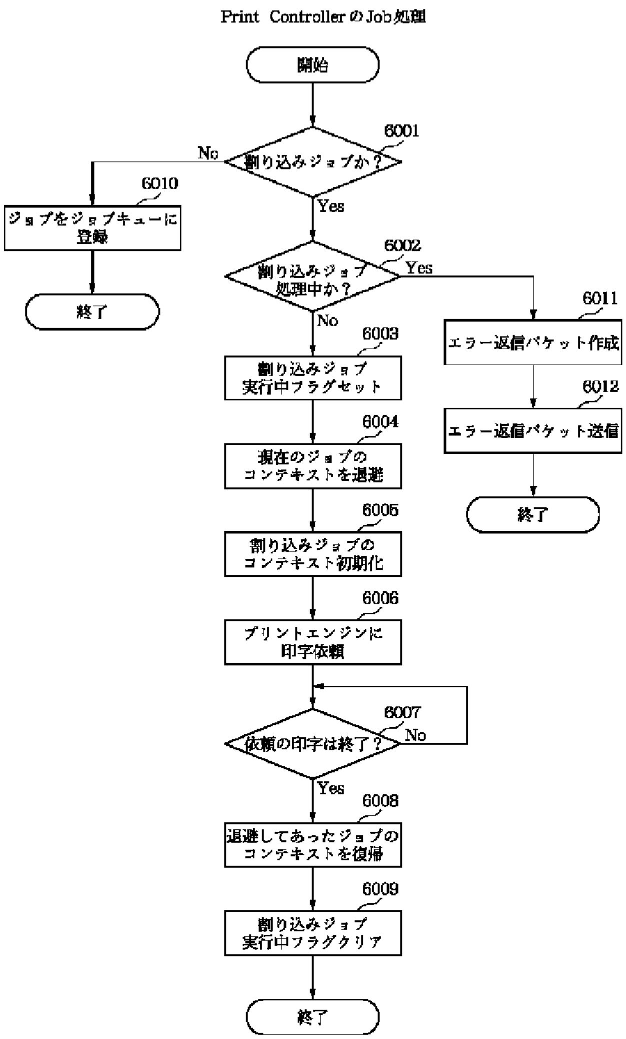
割込みプリントが完了しました！

● 終了時間優先
○ 画質優先
○ プリントエンジン指定

以下のプリンタに分散出力されました

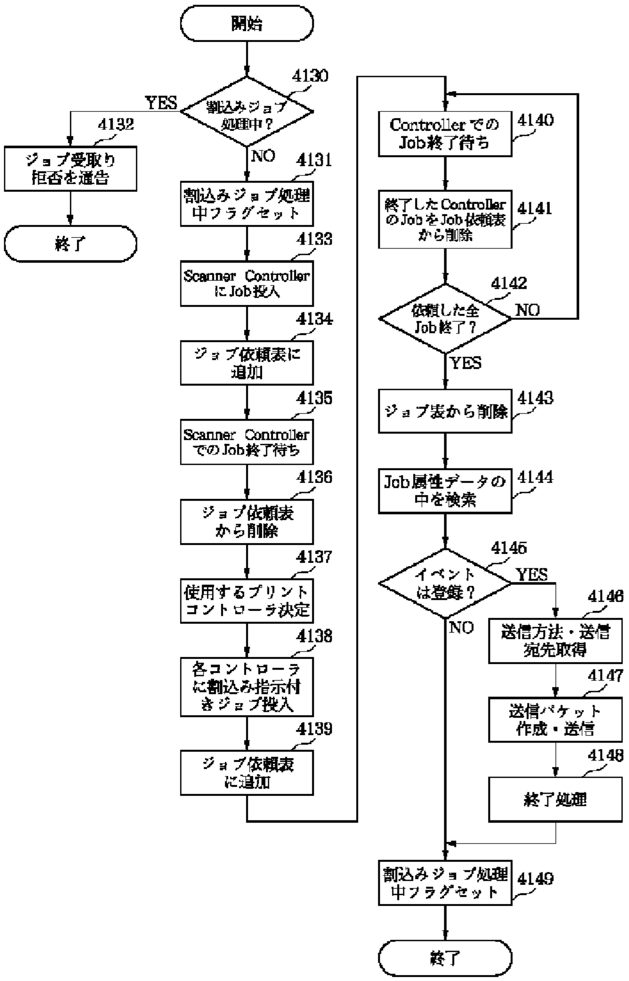
LBP1（フィニッシャ付き）…25ページ
LBP2（フィニッシャ無し）…20ページ
インクジェットプリンタ　… 5ページ

[Drawing 90]

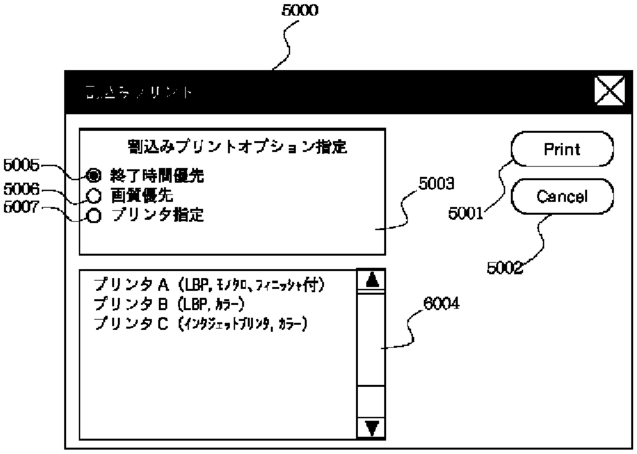

Print Controller のJob処理
開始
6001
割り込みジョブか？
No
Yes
6010
ジョブをジョブキューに登録
終了
6002
割り込みジョブ処理中か？
Yes
No
6011
エラー返信パケット作成
6012
エラー返信パケット送信
終了
6003
割り込みジョブ実行中フラグセット
6004
現在のジョブのコンテキストを退避
6005
割り込みジョブのコンテキスト初期化
6006
プリントエンジンに印字依頼
6007
依頼の印字は終了？
No
Yes
6008
退避してあったジョブのコンテキストを復帰
6009
割り込みジョブ実行中フラグクリア
終了

[Drawing 91]

割り込みコピージョブマネージャでのジョブ処理



[Drawing 98]

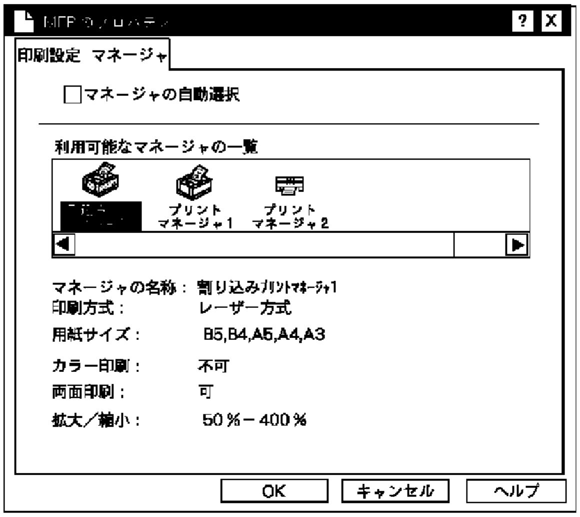


割込みプリントオプション指定

● 終了時間優先
○ 画質優先
○ プリンタ指定

プリンタ A（LBP, モノクロ、フィニッシャ付）
プリンタ B（LBP, カラー）
プリンタ C（インクジェットプリンタ, カラー）

[Drawing 99]



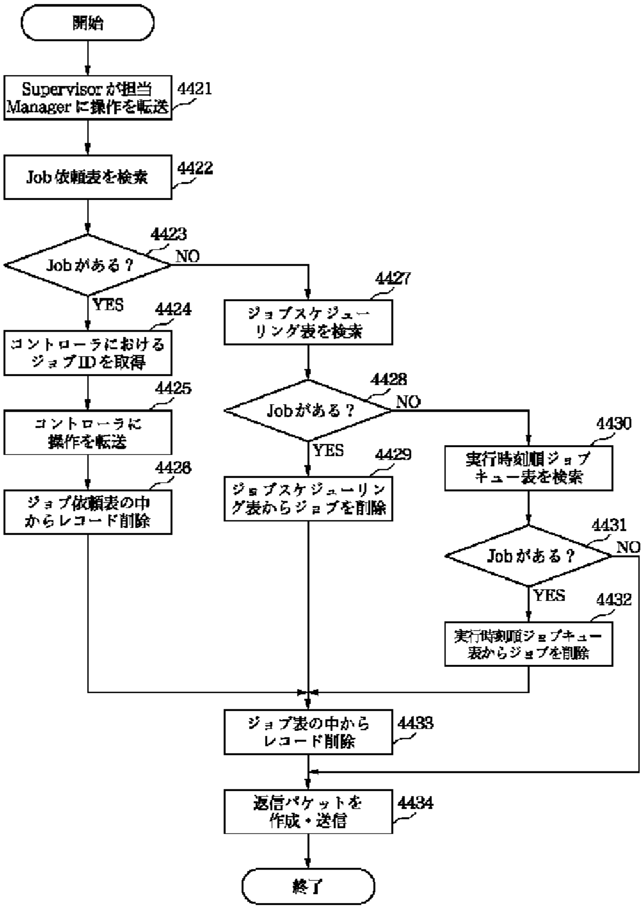
□マネージャの自動選択

利用可能なマネージャの一覧

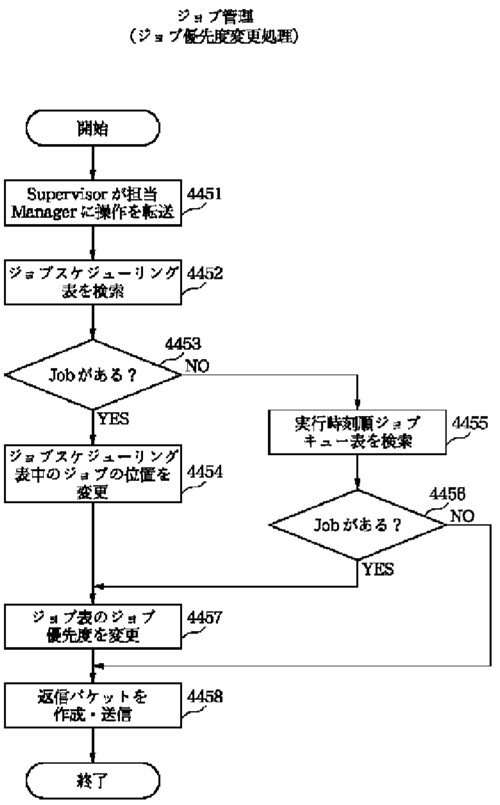
マネージャの名称： 割り込みﾌﾟﾘﾝﾄﾏﾈｰｼﾞｬ1
印刷方式： レーザー方式
用紙サイズ： B5,B4,A5,A4,A3
カラー印刷： 不可
両面印刷： 可
拡大／縮小： 50％－400％

[Drawing 92]

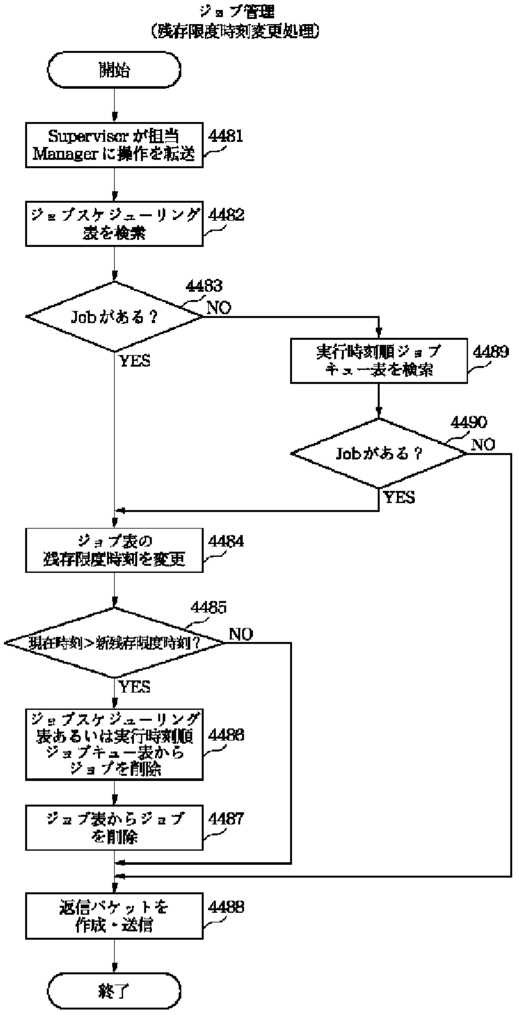
各マネージャでのジョブ管理
（ジョブ削除処理）


開始
Supervisorが担当Managerに操作を転送
4421
Job依頼表を検索
4422
Jobがある？
4423
NO
YES
コントローラにおけるジョブIDを取得
4424
コントローラに操作を転送
4425
ジョブ依頼表の中からレコード削除
4426
ジョブスケジューリング表を検索
4427
Jobがある？
4428
NO
YES
ジョブスケジューリング表からジョブを削除
4429
実行時刻順ジョブキュー表を検索
4430
Jobがある？
4431
NO
YES
実行時刻順ジョブキュー表からジョブを削除
4432
ジョブ表の中からレコード削除
4433
返信パケットを作成・送信
4434
終了

[Drawing 93]


ジョブ管理
（ジョブ優先度変更処理）
開始
Supervisor が担当 Manager に操作を転送
4451
ジョブスケジューリング表を検索
4452
Job がある？
4453
NO
YES
実行時刻順ジョブキュー表を検索
4455
ジョブスケジューリング表中のジョブの位置を変更
4454
Job がある？
4456
NO
YES
ジョブ表のジョブ優先度を変更
4457
返信パケットを作成・送信
4458
終了

[Drawing 95]


ジョブ管理
（残存限度時刻変更処理）
開始
Supervisor が担当 Manager に操作を転送
4481
ジョブスケジューリング表を検索
4482
Job がある？
4483
NO
YES
実行時刻順ジョブキュー表を検索
4489
Job がある？
4490
NO
YES
ジョブ表の残存限度時刻を変更
4484
現在時刻＞新残存限度時刻？
4485
NO
YES
ジョブスケジューリング表あるいは実行時刻順ジョブキュー表からジョブを削除
4486
ジョブ表からジョブを削除
4487
返信パケットを作成・送信
4488
終了

[Translation done.]